

So3p1179

日 本 国 特 許 庁
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office.

出 願 年 月 日 2 0 0 2 年 1 0 月 1 日
Date of Application:

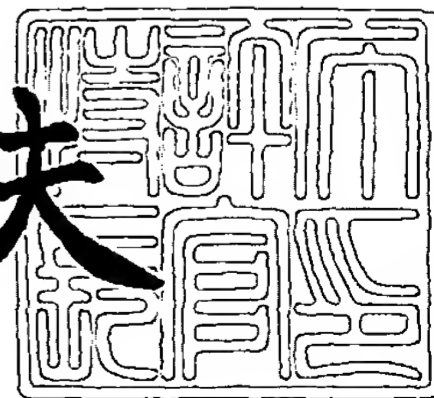
出 願 番 号 特 願 2 0 0 2 - 2 8 9 0 2 2
Application Number:
[ST. 10/C]: [J P 2 0 0 2 - 2 8 9 0 2 2]

出 願 人 ソニー株式会社
Applicant(s):

2 0 0 3 年 8 月 1 1 日

特 許 庁 長 官
Commissioner,
Japan Patent Office

今 井 康 夫



出 証 番 号 出 証 特 2 0 0 3 - 3 0 6 4 3 4 6

【書類名】 特許願

【整理番号】 0290283704

【提出日】 平成14年10月 1日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 7/00

【発明者】

 【住所又は居所】 東京都品川区北品川 6 丁目 7 番 3 5 号 ソニー株式会社
 内

 【氏名】 住広 博

【特許出願人】

 【識別番号】 000002185

 【氏名又は名称】 ソニー株式会社

【代理人】

 【識別番号】 100082131

 【弁理士】

 【氏名又は名称】 稲本 義雄

 【電話番号】 03-3369-6479

【手数料の表示】

 【予納台帳番号】 032089

 【納付金額】 21,000円

【提出物件の目録】

 【物件名】 明細書 1

 【物件名】 図面 1

 【物件名】 要約書 1

 【包括委任状番号】 9708842

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 情報処理装置および方法、記録媒体、並びにプログラム

【特許請求の範囲】

【請求項 1】 複数のモジュール毎にバスの取得を要求するリクエスト信号を受信する複数の受信手段と、

前記複数の受信手段毎に受信されたリクエスト信号に基づいて、前記複数のモジュール毎の待ち時間を計測する計測手段と、

前記計測手段により計測された待ち時間に応じて、前記複数のモジュールのバス取得の優先順位を決定する優先順位決定手段と、

前記優先順位決定手段により決定された優先順位に基づいて、前記複数のモジュールにバスを取得するように制御する制御手段と

を含むことを特徴とする情報処理装置。

【請求項 2】 前記計測手段により計測された前記待ち時間が同一である複数のモジュールが存在する場合、前記優先順位決定手段は、前記優先順位をラウンドロビン方式により決定する

ことを特徴とする請求項 1 に記載の情報処理装置。

【請求項 3】 複数のモジュール毎にバスの取得を要求するリクエスト信号を受信する複数の受信ステップと、

前記複数の受信ステップの処理毎に受信されたリクエスト信号に基づいて、前記複数のモジュール毎の待ち時間を計測する計測ステップと、

前記計測ステップの処理により計測された待ち時間に応じて、前記複数のモジュールのバス取得の優先順位を決定する優先順位決定ステップと、

前記優先順位決定ステップの処理で決定された優先順位に基づいて、前記複数のモジュールにバスを取得するように制御する制御ステップと

を含むことを特徴とする情報処理方法。

【請求項 4】 複数のモジュール毎に受信されるバスの取得を要求するリクエスト信号に基づいて、前記複数のモジュール毎の待ち時間を計測する計測ステップと、

前記計測ステップの処理により計測された待ち時間に応じて、前記複数のモジ

ジュールのバス取得の優先順位を決定する優先順位決定ステップと、

前記優先順位決定ステップの処理で決定された優先順位に基づいて、前記複数のモジュールにバスを取得するように制御する制御ステップと

を含むことを特徴とするコンピュータが読み取り可能なプログラムが記録されている記録媒体。

【請求項 5】 複数のモジュール毎に受信されるバスの取得を要求するリクエスト信号に基づいて、前記複数のモジュール毎の待ち時間を計測する計測ステップと、

前記計測ステップの処理により計測された待ち時間に応じて、前記複数のモジュールのバス取得の優先順位を決定する優先順位決定ステップと、

前記優先順位決定ステップの処理で決定された優先順位に基づいて、前記複数のモジュールにバスを取得するように制御する制御ステップと

を含むことを特徴とする処理をコンピュータに実行させるプログラム。

【発明の詳細な説明】

【 0 0 0 1 】

【発明の属する技術分野】

本発明は、情報処理装置および方法、記録媒体、並びにプログラムに関し、特に、複数のモジュールが1つのバスを利用した処理をする際に、そのバスを複数のモジュール間で効率よく取得できるように調停する情報処理装置および方法、記録媒体、並びにプログラムに関する。

【 0 0 0 2 】

【従来の技術】

大規模集積回路（LSI：Large Scale Integration）により制御される電子機器が一般に普及してきている。

【 0 0 0 3 】

LSIにおいては、複数のIC（Integrated Circuit）からなるモジュール（エンジン）が様々な処理を実行する。この際、LSIの各モジュールが処理を実行する上で、処理に応じて外部メモリに対してデータを記憶させ、または、記憶させたデータを適宜読み出す処理が必要となる。

【 0 0 0 4 】

このため、モジュール毎に、外部メモリと接続されたバスの取得順序を調停する、いわゆる、バスアービトレーションを制御する機能が設けられており、各モジュールのバスの取得を制御している。

【 0 0 0 5 】

バスアービトレーションの方法としては、ラウンドロビン方式や、固定プライオリティ（固定優先順位）方式が採用されている。

【 0 0 0 6 】

ラウンドロビン方式とは、複数のモジュールに対して均等にバスを取得できるようにさせる方式であり、換言すれば、複数のモジュールが順番にバスを取得できるようにするように制御する方式である。

【 0 0 0 7 】

図 1 A, B は、第 1 モジュールと第 2 モジュールの 2 個のモジュールが、バスを取得するタイミングを、ラウンドロビン方式で制御するときのタイミングチャートの例を示している。

【 0 0 0 8 】

ここで、図 1 A は、第 1 モジュールがバスの取得を要求するリクエスト信号を送信するタイミングと、バスを取得して所定の第 1 の処理を実行するタイミングを示しており、図 1 B は、第 2 モジュールがバスの取得を要求するリクエスト信号を送信するタイミングと、バスを取得して所定の第 2 の処理を実行するタイミングを示している。

【 0 0 0 9 】

第 1 モジュールは、時刻 t_1 において、バスの取得を要求するリクエスト信号 R_1 を送信し、時刻 t_1 乃至 t_2 において、バスが取得されると、第 1 の処理 1（図中においては、○中に「1」を付して表示している：尚、以下、図中の処理を示す表記においては、○中に番号を付すものとする）を実行し、その処理が終了したタイミングの時刻 t_2 において、リクエスト信号 R_2 を送信する。このとき、ラウンドロビン方式では、第 1 モジュールと第 2 モジュールが交互にバスが取得されるように制御されるため、第 2 モジュールが第 2 の処理 1 1 を完了させ

るまで、第1モジュールはバスを取得することができない。そこで、その次の第1の処理2は、実行されず、待ちの状態となる。

【0 0 1 0】

ここで、時刻 t_3 において、第2モジュールはリクエスト信号 R_{11} を送信し、そのタイミングでは、第2モジュールがバスを取得し、時刻 t_3 乃至 t_4 において、第2の処理11を実行する。

【0 0 1 1】

この第2の処理11が終了する時刻 t_4 において、第1モジュールはバスを取得することができるので、時刻 t_4 乃至 t_5 において、第1の処理2を実行し、さらに、その第1の処理2が終了する時刻 t_5 において、バスの取得を要求するリクエスト信号 R_3 を送信する。

【0 0 1 2】

このときも、ラウンドロビン法においては、第1モジュールは、第2モジュールの処理が完了するまでバスを取得することができない。

【0 0 1 3】

この状態で、時刻 t_6 において、第2モジュールがバス取得のリクエスト信号 R_{12} を送信すると、そのタイミングでバスを取得することができるので、時刻 t_6 乃至 t_7 において、第2の処理12を実行する。

【0 0 1 4】

この処理12が終了する時刻 t_7 において、第1モジュールはバスを取得することができるので、時刻 t_7 乃至 t_8 において、第1の処理3を実行し、さらに、時刻 t_8 において、バスの取得を要求するリクエスト信号 R_4 を送信する。

【0 0 1 5】

このときも、第1モジュールは、ラウンドロビン法においては、第2モジュールの処理が完了するまでバスを取得することができない。

【0 0 1 6】

この状態で、時刻 t_9 において、第2モジュールがバス取得のリクエスト信号 R_{13} を送信すると、そのタイミングでバスを取得することができるので、時刻 t_9 乃至 t_{10} において、第2の処理13を実行し、さらに、このタイミングで

バスの取得を要求するリクエスト信号 R 1 4 を送信する。このとき、第 2 モジュールは、ラウンドロビン法において、第 1 モジュールの処理が完了するまでバスを取得することができない。

【 0 0 1 7 】

一方、この第 2 の処理 1 3 が実行されたタイミングである時刻 t 1 0 において、第 1 モジュールはバスを取得することができるので、時刻 t 1 0 乃至 t 1 1 において、第 1 の処理 4 を実行する。このとき、バス取得のリクエストは送信されない。

【 0 0 1 8 】

この第 1 の処理 4 が終了する時刻 t 1 1 において、第 2 モジュールはバスを取得することができるので、時刻 t 1 1 乃至 t 1 2 において、第 2 の処理 1 4 を実行し、さらに、時刻 t 1 2 において、バスの取得を要求するリクエスト信号 R 1 5 を送信する。このときも、第 2 モジュールは、ラウンドロビン法において、第 1 モジュールの処理が完了するまでバスを取得することができない。

【 0 0 1 9 】

この状態で、時刻 t 1 3 において、第 1 モジュールがバスの取得を要求するリクエスト信号 R 5 を送信すると、そのタイミングでバスを取得することができるので、時刻 t 1 3 乃至 t 1 4 において、第 1 の処理 5 を実行する。

【 0 0 2 0 】

この第 1 の処理 5 が終了する時刻 t 1 4 において、第 2 モジュールはバスを取得することができるので、時刻 t 1 4 乃至 t 1 5 において、第 2 の処理 1 5 を実行し、さらに、時刻 t 1 5 において、バスの取得を要求するリクエスト信号 R 1 6 を送信する。このときも、第 2 モジュールは、ラウンドロビン法において、第 1 モジュールの処理が完了するまでバスを取得することができない。

【 0 0 2 1 】

この状態で、時刻 t 1 6 において、第 1 モジュールがバスの取得を要求するリクエスト信号 R 6 を送信すると、そのタイミングでバスを取得することができるので、時刻 t 1 6 乃至 t 1 7 において、第 1 の処理 6 を実行する。

【 0 0 2 2 】

この第 1 の処理 6 が終了する時刻 t_{17} において、第 2 モジュールはバスを取得することができるので、時刻 t_{17} 乃至 t_{18} において、第 1 の処理 1 6 を実行する。

【 0 0 2 3 】

上述のように、ラウンドロビン方式とは、第 1 モジュールと第 2 モジュールが交互にバスを取得して処理を行う方式である。

【 0 0 2 4 】

また、固定プライオリティ方式（固定優先順位方式）とは、複数のモジュールが存在する場合に、そのモジュール毎に、優先順位を予め設定し、各モジュールからバスの取得を要求するリクエストがあった場合、その優先順位に応じてバスを取得させる方式である。

【 0 0 2 5 】

また、従来のアービトレーション装置には、1 つの共有資源を複数のマスタで使用する際に、優先度を算出して算出結果に基づいて共有資源の使用許可を設定するものがある（例えば、特許文献 1 参照）。

【 0 0 2 6 】

【特許文献 1】

特開 2 0 0 2 - 5 5 9 4 4 号公報（段落 [0 0 5 8] 、図 3 、図 4 ）

【 0 0 2 7 】

【発明が解決しようとする課題】

ところで、上述の図 1 A 、 B で示したように、各モジュールの外部メモリへのアクセスの頻度は、処理状態により変化するのが一般的である。

【 0 0 2 8 】

しかしながら、例えば、図 1 A 、 B においては、例えば、時刻 t_2 、 t_3 、 t_4 において、第 1 モジュールがバスの取得を要求するリクエスト信号 R_2 、 R_3 、 R_4 を送信しているのに対して、第 2 モジュールがバスの取得を要求していないにもかかわらず、第 2 モジュールの処理が終わるタイミングである時刻 t_4 、 t_7 、 t_{10} まで、その処理を待たなければならず、時刻 t_{12} 、 t_{15} において、第 2 モジュールがバスを取得するリクエスト R_{15} 、 R_{16} を送信している

のに対して、第 1 モジュールがバスの取得を要求していないにもかかわらず、第 1 モジュールの処理が終わるタイミングである時刻 t_{14} 、 t_{17} まで、その処理を待たなければならないため、効率のよいバスアービトレーション処理（バスの取得を調整する処理）が実現できず、結果として無駄な待ち時間が発生し、処理速度が LSI 全体として低減してしまうという課題があった。

【 0 0 2 9 】

また、固定優先順位方式においては、多くのモジュールが存在する場合、優先順位の高いモジュールは、優先的にバスを取得して処理を実行させることができるが、優先順位の低いモジュールは、優先順位の高い処理が実行され続けている限り、バスを取得することができないことになってしまうため、モジュール毎の処理状態が変化したときに適正なバスの取得ができない恐れがあり、結果として処理速度を低下させてしまう恐れがあるという課題があった。

【 0 0 3 0 】

本発明はこのような状況に鑑みてなされたものであり、優先順位を変化させて適正なバスアービトレーション処理を実現させるものである。

【 0 0 3 1 】

【課題を解決するための手段】

本発明の情報処理装置は、複数のモジュール毎にバスの取得を要求するリクエスト信号を受信する複数の受信手段と、複数の受信手段毎に受信されたリクエスト信号に基づいて、複数のモジュール毎の待ち時間を計測する計測手段と、計測手段により計測された待ち時間に応じて、複数のモジュールのバス取得の優先順位を決定する優先順位決定手段と、優先順位決定手段により決定された優先順位に基づいて、複数のモジュールにバスを取得するように制御する制御手段とを含むことを特徴とする。

【 0 0 3 2 】

前記計測手段により計測された待ち時間が同一である複数のモジュールが存在する場合、優先順位決定手段には、優先順位をラウンドロビン方式により決定させるようにすることができる。

【 0 0 3 3 】

本発明の情報処理方法は、複数のモジュール毎にバスの取得を要求するリクエスト信号を受信する複数の受信ステップと、複数の受信ステップの処理毎に受信されたリクエスト信号に基づいて、複数のモジュール毎の待ち時間を計測する計測ステップと、計測ステップの処理により計測された待ち時間に応じて、複数のモジュールのバス取得の優先順位を決定する優先順位決定ステップと、優先順位決定ステップの処理で決定された優先順位に基づいて、複数のモジュールにバスを取得するように制御する制御ステップとを含むことを特徴とする。

【 0 0 3 4 】

本発明の記録媒体のプログラムは、複数のモジュール毎に受信されるバスの取得を要求するリクエスト信号に基づいて、複数のモジュール毎の待ち時間を計測する計測ステップと、計測ステップの処理により計測された待ち時間に応じて、複数のモジュールのバス取得の優先順位を決定する優先順位決定ステップと、優先順位決定ステップの処理で決定された優先順位に基づいて、複数のモジュールにバスを取得するように制御する制御ステップとを含むことを特徴とする。

【 0 0 3 5 】

本発明のプログラムは、複数のモジュール毎に受信されるバスの取得を要求するリクエスト信号に基づいて、複数のモジュール毎の待ち時間を計測する計測ステップと、計測ステップの処理により計測された待ち時間に応じて、複数のモジュールのバス取得の優先順位を決定する優先順位決定ステップと、優先順位決定ステップの処理で決定された優先順位に基づいて、複数のモジュールにバスを取得するように制御する制御ステップとを含むことを特徴とする。

【 0 0 3 6 】

本発明の情報処理装置および方法、並びにプログラムにおいては、複数のモジュール毎にバスの取得を要求するリクエスト信号が受信され、複数の受信されたリクエスト信号に基づいて、複数のモジュール毎の待ち時間が計測され、計測された待ち時間に応じて、複数のモジュールのバス取得の優先順位が決定され、決定された優先順位に基づいて、複数のモジュールにバスが取得されるように制御される。

【 0 0 3 7 】

【発明の実施の形態】

図 2 は、本発明に係る LSI の一実施の形態の構成を示す図である。

【 0 0 3 8 】

本発明における LSI は、入力画像を MPEG 方式で符号化して出力すると共に、入力された MPEG 方式の符号化信号を復号して画像として出力する処理を同時に実行することができるものである。

【 0 0 3 9 】

まず、符号化部 1 の構成について説明する。符号化部 1 は、入力画像を符号化し、符号化信号として出力するものである。

【 0 0 4 0 】

符号化部 1 の画像入力部 2 1 は、入力されてくる画像を符号化で用いる空間解像度に変換し、I ピクチャ（イントラ画像）、P ピクチャ（前方向予測画像）、および、B ピクチャ（前後方向予測画像）のそれぞれのピクチャタイプに合わせて画面を並べ替えて、動き予測部 2 2、および、符号化処理部 2 3 に出力する。より詳細には、画像入力部 2 1 は、ピクチャタイプ毎にバス 8 を介して一旦外部メモリ 9 に出力して、記憶させた後、これを動き予測部 2 2、および、符号化処理部 2 3 が読み出す。このとき、画像入力部 2 1 は、符号化制御部 2 5 に対してバス 8 を取得するように要求し、この要求に応じて、符号化制御部 2 5 からバス 8 が取得されたことを示す確認の通知を受信すると、バス 8 を介して外部メモリ 9 に対して所定の処理を実行させる。

【 0 0 4 1 】

動き予測部 2 2 は、B ピクチャ、または、P ピクチャのピクチャタイプとして入力された画像と、その基準となる I ピクチャ、または、P ピクチャとの差分を求め、動きベクトルを求めて符号化処理部 2 3 に出力する。より詳細には、動き予測部 2 2 は、外部メモリ 9 に入力画像として記憶されている画像と相関のある、I ピクチャ、または、P ピクチャを外部メモリ 9 から読み出して、比較し、動きベクトルを求めて符号化処理部 2 3 が読み出せるように、外部メモリ 9 に記憶させる。このとき、動き予測部 2 2 は、符号化制御部 2 5 に対してバス 8 を取得するように要求し、この要求に応じて、符号化制御部 2 5 からバス 8 が取得されたこ

とを示す確認の通知を受信すると、バス 8 を介して外部メモリ 9 に対して所定の処理を実行させる。

【 0 0 4 2 】

符号化処理部 2 3 は、画像入力部 2 1 より入力された入力画像と、動き予測部 2 2 より入力された動きベクトルに基づいて、ピクチャタイプに応じて符号化信号を生成してビットストリーム出力部 2 4 に出力する。より詳細には、外部メモリ 9 に記憶されている画像入力部 2 1 より入力された入力画像と、動き予測部 2 2 により生成された動きベクトルとを用いて、Iピクチャの場合、入力画像となった画像をそのまま符号化し、Pピクチャ、または、Bピクチャの場合、その動きベクトルを符号化することにより符号化信号を生成してビットストリーム出力部 2 4 により読出し可能な情報として、外部メモリ 9 に記憶させる。このとき、符号化処理部 2 3 は、符号化制御部 2 5 に対してバス 8 を取得するように要求し、この要求に応じて、符号化制御部 2 5 からバス 8 が取得されたことを示す確認の通知を受信すると、バス 8 を介して外部メモリ 9 に対して所定の処理を実行させる。

【 0 0 4 3 】

ビットストリーム出力部 2 4 は、符号化処理部 2 3 により符号化された符号化信号を外部メモリ 9 から読み出し、ビットストリームとして符号化出力信号を出力する。このとき、ビットストリーム出力部 2 4 は、符号化制御部 2 5 に対してバス 8 を取得するように要求し、この要求に応じて、符号化制御部 2 5 からバス 8 が取得されたことを示す確認の通知を受信すると、バス 8 を介して外部メモリ 9 に対して所定の処理を実行させる。

【 0 0 4 4 】

符号化制御部 2 5 は、符号化部 1 の全体の動作を制御しており、画像入力部 2 1、動き予測部 2 2、符号化処理部 2 3、またはビットストリーム出力部 2 4 からバス 8 を取得するように要求があった場合、バスの取得を要求するリクエスト信号 (Request) を端子 2 5 a からカウンタ 3 の端子 3 a およびアービトレーションコントローラ 7 の端子 7 e に出力する。また、このリクエスト信号に応じてバス 8 の取得が認められたとき、アービトレーションコントローラ 7 の端子 7 d

から送信されてくる確認信号 (Acknowledge) を端子 2 5 b で受信すると共に、この確認信号が受信されたとき、画像入力部 2 1、動き予測部 2 2、符号化処理部 2 3、またはビットストリーム出力部 2 4 に対して、バス 8 が取得されたことを示す通知を出力して、各種の処理を実行させる。

【0 0 4 5】

カウンタ 3 は、端子 3 a に符号化部 1 より入力されるバス 8 の取得を要求するリクエスト信号を受信すると、所定の時間間隔で値をカウントし、そのカウント値 Z 1 をコンパレータ 4 - 1, 4 - 2 に出力する。また、カウンタ 3 は、アービトレーションコントローラ 7 の端子 7 d から端子 3 b を介して確認信号が入力されると、それまでのカウントを停止し、カウント値をリセットする。尚、この状態でも、カウンタ 3 は、カウント値をコンパレータ 4 - 1, 4 - 2 に出力している。

【0 0 4 6】

コンパレータ 4 - 1, 4 - 2 は、それぞれカウンタ 3 から入力されたカウント値 Z 1 と、所定の値 C 1, C 2 とを比較して、カウント値 Z 1 の方が大きいと判定するとき、True の信号をアービトレーションコントローラ 7 の端子 7 b, 7 c にそれぞれ出力し、それ以外るとき False の信号をアービトレーションコントローラ 7 の端子 7 b, 7 c にそれぞれ出力する。尚、所定の値 C 1, C 2 は、 $C 1 \neq C 2$ であるため、例えば、 $C 1 < C 2$ である場合、カウンタ 3 のカウント値が継続されている状態のとき、まず、カウント値 Z 1 が所定の値 C 1 を越えたタイミングで、コンパレータ 4 - 1 から True の信号が出力され、さらに、カウント値 Z 1 が所定の値 C 2 を越えたタイミングで、コンパレータ 4 - 2 からも True の信号が出力されることになる。従って、アービトレーションコントローラ 7 は、コンパレータ 4 - 1, 4 - 2 から端子 7 b, 7 c に入力される信号のパターン（組み合わせ）により符号化部 1 がバス 8 の取得を要求するリクエスト信号が送信されてから、バス 8 の取得を確認する確認信号が入力されるまでの時間を把握することができる。尚、以下の説明においては、 $C 1 < C 2$ であるものとするが、もちろん、 $C 1 > C 2$ であってもよい。

【0 0 4 7】

次に、復号部 2 の構成について説明する。

【 0 0 4 8 】

復号部 2 のビットストリーム入力部 4 2 は、入力符号化信号をバス 8 を介して外部メモリ 9 に順次記憶させる。このとき、ビットストリーム入力部 4 2 は、復号制御部 4 1 に対してバス 8 を取得するように要求し、この要求に応じて、復号制御部 4 1 からバス 8 が取得されたことを示す確認の通知を受信すると、バス 8 を介して外部メモリ 9 に対して所定の処理を実行させる。

【 0 0 4 9 】

復号処理部 4 3 は、ビットストリーム入力部 4 2 から順次外部メモリ 9 に記憶された符号化信号を読み出し、動き補償部 4 4 に出力する。このとき、復号処理部 4 3 は、一度復号した I ピクチャ、または、P ピクチャを外部メモリ 9 に記憶させた後、P ピクチャや B ピクチャの予測に用いるため再び外部メモリ 9 から読出す。また、復号処理部 4 3 は、入力符号化信号に含まれている予測モードや動きベクトルの情報を動き補償部 4 4 に出力し、この情報に基づいて、動き補償部 4 4 により生成された動き補償予測情報を用いて画像を復号する。このとき、復号処理部 4 3 は、復号制御部 4 1 に対してバス 8 を取得するように要求し、この要求に応じて、復号制御部 4 1 からバス 8 が取得されたことを示す確認の通知を受信すると、バス 8 を介して外部メモリ 9 に対して所定の処理を実行させる。

【 0 0 5 0 】

動き補償部 4 4 は、復号処理部 4 3 から入力される復号された画像、動きベクトル、および、予測モードの情報に基づいて、予測モードが動き補償予測モードであるとき、動き補償予測情報を生成して復号された画像に動き補償予測を行い、復号処理部 4 3 に出力すると共に、復号処理部 4 3 から入力された画像を画像出力部 4 5 に出力する。より詳細には、復号処理部 4 3 から入力された画像を一旦バス 8 を介して外部メモリ 9 に記憶させて、外部出力部 4 5 から読み出せるように記憶させる。このとき、動き補償部 4 4 は、復号制御部 4 1 に対してバス 8 を取得するように要求し、この要求に応じて、復号制御部 4 1 からバス 8 が取得されたことを示す確認の通知を受信すると、バス 8 を介して外部メモリ 9 に対して所定の処理を実行させる。

【 0 0 5 1 】

画像出力部 4 5 は、バス 8 を介して外部メモリ 9 に記憶されている復号された画像を順次読み出して、出力画像として出力する。このとき、画像出力部 4 5 は、復号制御部 4 1 に対してバス 8 を取得するように要求し、この要求に応じて、復号制御部 4 1 からバス 8 が取得されたことを示す確認の通知を受信すると、バス 8 を介して外部メモリ 9 に対して所定の処理を実行させる。

【 0 0 5 2 】

復号制御部 4 1 は、復号部 2 の全体の動作を制御しており、ビットストリーム入力部 4 2、復号処理部 4 3、動き補償部 4 4、または画像出力部 4 5 からバス 8 を取得するように要求があった場合、バスの取得を要求するリクエスト信号 (Request) を端子 4 1 a からカウンタ 5 の端子 5 a およびアービトレーションコントローラ 7 の端子 7 g に出力する。また、このリクエストに応じてバス 8 の取得が認められたとき、アービトレーションコントローラ 7 の端子 7 f から送信されてくる確認信号 (Acknowledge) を端子 4 1 b で受信すると共に、この確認信号が受信されたとき、ビットストリーム入力部 4 2、復号処理部 4 3、動き補償部 4 4、または画像出力部 4 5 に対して、バス 8 が取得されたことを示す通知を出力して、各種の処理を実行させる。

【 0 0 5 3 】

カウンタ 5 は、カウンタ 3 と同様のものである。すなわち、カウンタ 5 は、端子 5 a に復号部 2 より入力されるバス 8 の取得を要求するリクエスト信号を受信すると、所定の時間間隔で値をカウントし、そのカウント値 Z 2 をコンパレータ 6 - 1, 6 - 2 に出力する。また、カウンタ 5 は、アービトレーションコントローラ 7 から端子 5 b を介して確認信号が入力されると、それまでカウントを停止して、カウント値 Z 2 をリセットする。尚、この状態でも、カウンタ 3 は、カウント値をコンパレータ 6 - 1, 6 - 2 に出力している。

【 0 0 5 4 】

コンパレータ 6 - 1, 6 - 2 は、コンパレータ 4 - 1, 4 - 2 と同様のものである。すなわち、コンパレータ 6 - 1, 6 - 2 は、それぞれカウンタ 5 から入力されたカウント値 Z 2 と、所定の値 C 1, C 2 とを比較して、カウント値 Z 2 の

方が大きいと判定するとき、Trueの信号をアービトレーションコントローラ7の端子7 h, 7 i にそれぞれ出力し、それ以外るときFalseの信号をアービトレーションコントローラ7の端子7 h, 7 i にそれぞれ出力する。尚、所定の値C 1, C 2 は、 $C 1 \neq C 2$ であるため、例えば、 $C 1 < C 2$ である場合、カウンタ5のカウンタ値Z 2 が継続されている状態のとき、まず、カウンタ値Z 2 が所定の値C 1 を越えたタイミングで、コンパレータ6 - 1 からTrueの信号が出力され、さらに、カウンタ値Z 2 が所定の値C 2 を越えたタイミングで、コンパレータ6 - 2 からTrueの信号が出力されることになる。従って、アービトレーションコントローラ7は、端子7 h, 7 i に入力されるコンパレータ6 - 1, 6 - 2 から入力される信号のパターンにより復号部2がバス8の取得を要求するリクエスト信号を送信してから、バス8の取得を確認する確認信号が入力されるまでの時間を把握することができる。

【0 0 5 5】

アービトレーションコントローラ7は、符号化部1、および、復号部2から端子7 e, 7 g を介して入力されるバス8の取得を要求するリクエスト信号、並びに、端子7 b, 7 c, 7 h、および7 i から入力されるコンパレータ4 - 1, 4 - 2, 6 - 1, 6 - 2 のそれぞれの判定結果に基づいて、符号化部1、および、復号部2のいずれにバス8の取得させるかを調整し、取得を認めるとき、確認信号を端子7 d, 7 f より出力すると共に、確認信号を送信した符号化部1、または、復号部2のいずれかのモジュールに対してバス8を取得させるようにし、外部メモリ9へのデータの書込みと、読出しを可能な状態にする。また、アービトレーションコントローラ7は、メモリ7 a を内蔵しており、バス8の取得を認めたモジュールの情報を記憶し、ラウンドロビン方式によりバス8の取得を決定する際にその情報を利用し、直前にバス8を取得したタイミングが最も古いモジュールがバス8を取得できるように調整する。すなわち、モジュールが、符号化部1と復号部2の2個の場合、直前にバス8を取得したモジュールが符号化部1であったときには、復号部2がバス8を取得できるように調整し、逆に、復号部2が直前にバスを取得していたとき、符号化部1がバス8を取得できるように調整する。

【 0 0 5 6 】

尚、図 2 において、太線はデータの流れを示すものであり、一点鎖線は、符号化部 1 によるデータをアービトレーションコントローラ 7 がバス 8 を介して外部メモリ 9 に授受する経路を示しており、2 点鎖線は、復号部 2 によるデータをアービトレーションコントローラ 7 がバス 8 を介して外部メモリ 9 に授受する経路を示している。

【 0 0 5 7 】

次に、図 3 のフローチャートを参照して、符号化制御部 2 5 による符号化制御処理について説明する。

【 0 0 5 8 】

ステップ S 1 において、符号化制御部 2 5 は、画像入力部 2 1、動き予測部 2 2、符号化部 2 3、または、ビットストリーム出力部 2 4 のいずれかからバス 8 の取得が要求されているか否かを判定し、画像入力部 2 1、動き予測部 2 2、符号化部 2 3、または、ビットストリーム出力部 2 4 のいずれかからバス 8 の取得が要求されるまでその処理を繰り返す。ステップ S 1 において、画像入力部 2 1、動き予測部 2 2、符号化部 2 3、または、ビットストリーム出力部 2 4 のいずれかからバス 8 の取得が要求された場合、バス 8 の取得が要求されたと判定され、ステップ S 2 において、符号化制御部 2 5 は、端子 2 5 a からカウンタ 3 の端子 3 a、および、アービトレーションコントローラ 7 の端子 7 e にバス 8 の取得を要求するリクエスト信号を送信する。

【 0 0 5 9 】

ステップ S 3 において、符号化制御部 2 5 は、アービトレーションコントローラ 7 の端子 7 d から自らの端子 2 5 b に確認信号が送信されてきたか否かを判定し、確認信号が送信されてくるまで、その処理を繰り返し、例えば、確認信号が送信されてきた、すなわち、バス 8 の取得が認められたと判定した場合、その処理は、ステップ S 4 に進む。

【 0 0 6 0 】

ステップ S 4 において、符号化制御部 2 5 は、バス 8 の取得を要求してきた、画像入力部 2 1、動き予測部 2 2、符号化部 2 3、または、ビットストリーム出

力部 2 4 のいずれかに対して、バス 8 の取得が認められたことを示す信号を送信し、各種の処理を実行させると共に、その処理は、ステップ S 1 に戻り、それ以降の処理が繰り返される。

【 0 0 6 1 】

次に、図 4 のフローチャートを参照して、カウンタ 3 によるカウント処理について説明する。

【 0 0 6 2 】

ステップ S 1 1 において、カウンタ 3 は、リクエスト信号が端子 3 a に入力されたか否かを判定する。例えば、図 3 のフローチャートを参照して説明したステップ S 2 の処理により、リクエスト信号が送信されてきた場合、リクエスト信号が端子 3 a に入力されたと判定し、ステップ S 1 2 において、カウンタ 3 は、カウントを開始する。

【 0 0 6 3 】

ステップ S 1 3 において、カウンタ 3 は、今現在カウントしているカウント値 Z 1 をコンパレータ 4 - 1, 4 - 2 に出力する。

【 0 0 6 4 】

ステップ S 1 4 において、カウンタ 3 は、端子 3 b にアービトレーションコントローラ 7 の端子 7 d から確認信号が入力されたか否かを判定し、確認信号が入力されていないと判定した場合、その処理は、ステップ S 1 3 に戻り、再び、カウント中のカウント値 Z 1 をコンパレータ 4 - 1, 4 - 2 に出力する。すなわち、確認信号が入力されるまで、ステップ S 1 3, S 1 4 の処理が繰り返し替えされ、時々刻々と変化して、カウントされていくカウント値 Z 1 がコンパレータ 4 - 1, 4 - 2 に出力され続ける。

【 0 0 6 5 】

ステップ S 1 4 において、アービトレーションコントローラ 7 の端子 7 d から確認信号が入力されたと判定された場合、ステップ S 1 5 において、カウンタ 3 は、それまで続けていたカウントを停止すると共に、カウント値 Z 1 をクリアして ($Z 1 = 0$ にして) 出力し、その処理は、ステップ S 1 1 に戻る。

【 0 0 6 6 】

すなわち、カウンタ 3 は、符号化部 1 からリクエスト信号が出力されてから、これに対応してアービトレーションコントローラ 7 から確認信号が入力されるまでの間、カウント値 Z 1 をカウントアップし続け、確認信号が入力されるとカウントを停止して、カウント値 Z 1 をクリア（リセット）する。

【 0 0 6 7 】

次に、図 5 のフローチャートを参照して、コンパレータ 4 - 1 による比較処理について説明する。

【 0 0 6 8 】

ステップ S 2 1 において、コンパレータ 4 - 1 は、カウンタ 3 より入力されるカウント値 Z 1 を取得する。

【 0 0 6 9 】

ステップ S 2 2 において、コンパレータ 4 - 1 は、カウンタ 3 より取得したカウント値 Z 1 と所定の値 C 1 とを比較し、カウント値 $Z 1 > C 1$ であるか否かを判定する。例えば、カウント値 Z 1 のカウントが進んでおらず、 $Z 1 > C 1$ ではないと判定された場合、ステップ S 2 3 において、コンパレータ 4 - 1 は、false の信号をアービトレーションコントローラ 7 の端子 7 b に出力し、その処理は、ステップ S 2 1 に戻る。

【 0 0 7 0 】

ステップ S 2 2 において、 $Z 1 > C 1$ であると判定された場合、ステップ S 2 4 において、コンパレータ 4 - 1 は、True の信号をアービトレーションコントローラ 7 の端子 7 b に出力し、その処理は、ステップ S 2 1 に戻り、それ以降の処理が繰り返される。

【 0 0 7 1 】

すなわち、カウンタ 3 のカウント値 Z 1 が、所定の値 C 1 を超えるまでカウントが進んだとき、すなわち、リクエスト信号が送信されたタイミングから、アービトレーションコントローラ 7 から確認信号が送信されてくるまでの待ち時間が所定の値 C 1 に対応する時間を超えたとき、アービトレーションコントローラ 7 の端子 7 b に True 信号が入力されることになる。

【 0 0 7 2 】

同様にして、コンパレータ 4 - 2 においても同様の処理がなされ、True、または、Falseの信号が、アービトレーションコントローラ 7 の端子 7 c に入力される。ただし、所定の値 C 1, C 2 は、 $C 1 \neq C 2$ であるので、例えば、 $C 1 < C 2$ の場合、カウント値 Z 1 の値が C 1 を超えた直後は、コンパレータ 4 - 1 から True 信号が出力されるが、コンパレータ 4 - 2 からは False 信号が出力されることになる。さらに、待ち時間が進み、カウント値 Z 1 が所定の値 C 2 を超えると、コンパレータ 4 - 1, 4 - 2 のいずれからも True 信号が出力されることになる。結果として、アービトレーションコントローラ 7 には、カウント値 Z 1 の進捗に応じて、コンパレータ 4 - 1, 4 - 2 から入力される True、または、false の組み合わせが変化して入力される。

【 0 0 7 3 】

次に、図 6 のフローチャートを参照して、復号部 2 の復号制御部 4 1 による復号制御処理について説明する。

【 0 0 7 4 】

ステップ S 3 1 において、復号制御部 4 1 は、ビットストリーム入力部 4 2、復号処理部 4 3、動き補償部 4 4、または、画像出力部 4 5 のいずれかからバス 8 の取得が要求されているか否かを判定し、復号制御部 4 1 は、ビットストリーム入力部 4 2、復号処理部 4 3、動き補償部 4 4、または、画像出力部 4 5 のいずれかからバス 8 の取得が要求されるまでその処理を繰り返す。ステップ S 3 1 において、復号制御部 4 1 は、ビットストリーム入力部 4 2、復号処理部 4 3、動き補償部 4 4、または、画像出力部 4 5 のいずれかからバス 8 の取得が要求された場合、バス 8 の取得が要求されたと判定され、ステップ S 3 2 において、復号制御部 4 1 は、端子 4 1 a からカウンタ 5 の端子 5 a、および、アービトレーションコントローラ 7 の端子 7 g にバス 8 の取得を要求するリクエスト信号を送信する。

【 0 0 7 5 】

ステップ S 3 3 において、符号化制御部 2 5 は、アービトレーションコントローラ 7 の端子 7 f から自らの端子 4 1 b に確認信号が送信されてきたか否かを判定し、確認信号が送信されてくるまで、その処理を繰り返し、例えば、確認信号

が送信されてきた、すなわち、バス 8 の取得が認められたと判定した場合、その処理は、ステップ S 3 4 に進む。

【 0 0 7 6 】

ステップ S 3 4 において、復号制御部 4 1 は、バス 8 の取得を要求してきたビットストリーム入力部 4 2、復号処理部 4 3、動き補償部 4 4、または、画像出力部 4 5 のいずれかに対して、バス 8 の取得が認められたことを示す信号を送信し、各種の処理を実行させると共に、その処理は、ステップ S 3 1 に戻り、それ以降の処理が繰り返される。

【 0 0 7 7 】

尚、カウンタ 5 によるカウント処理、および、コンパレータ 6 - 1， 6 - 2 による比較処理は、図 4 を参照して説明したカウンタ 3 によるカウント処理、および、図 5 を参照して説明して比較処理と同様であるので、その説明は省略する。ただし、カウンタ 5 によるカウント値 Z 2 は、コンパレータ 6 - 1， 6 - 2 にそれぞれ入力され、コンパレータ 6 - 1， 6 - 2 の比較結果となる信号は、アービトレーションコントローラ 7 の端子 7 h， 7 i に入力される。

【 0 0 7 8 】

次に、図 7 のフローチャートを参照して、アービトレーションコントローラ 7 による調停処理について説明する。

【 0 0 7 9 】

ステップ S 5 1 において、アービトレーションコントローラ 7 は、符号化部 1 のみからリクエスト信号を受信しているか否かを判定する。すなわち、アービトレーションコントローラ 7 は、端子 7 e に符号化部 1 からリクエスト信号が入力され、かつ、復号部 2 からリクエスト信号が入力されていない状態であるか否かを判定する。例えば、ステップ S 5 1 において、符号化部 1 からのみのリクエスト信号はないと判定された場合、その処理は、ステップ S 5 2 に進む。

【 0 0 8 0 】

ステップ S 5 2 において、アービトレーションコントローラ 7 は、復号部 2 のみからリクエスト信号を受信しているか否かを判定する。すなわち、アービトレーションコントローラ 7 は、端子 7 g に復号部 2 からリクエスト信号が入力され

、かつ、符号化部 1 からリクエスト信号が入力されていない状態であるか否かを判定する。例えば、ステップ S 5 2 において、復号部 2 からのみのリクエスト信号はないと判定された場合、その処理は、ステップ S 5 3 に進む。

【 0 0 8 1 】

ステップ S 5 3 において、アービトレーションコントローラ 7 は、符号化部 1、および、復号部 2 のいずれからもリクエスト信号が受信されているか否かを判定する。すなわち、アービトレーションコントローラ 7 は、端子 7 e に符号化部 1 からリクエスト信号が入力され、かつ、復号部 2 からリクエスト信号が入力されている状態であるか否かを判定する。例えば、ステップ S 5 3 において、符号化部 1、および、復号部 2 のいずれからもリクエスト信号はないと判定された場合、その処理は、ステップ S 5 1 に戻り、それ以降の処理が繰り返される。リクエスト信号は、符号化部 1、および、復号部 2 のいずれのモジュールからも入力されていないので、その処理は、ステップ S 5 1 乃至 S 5 3 の処理を繰り返す。

【 0 0 8 2 】

ステップ S 5 3 において、符号化部 1、および、復号部 2 のいずれのモジュールからもリクエスト信号が入力されていると判定された場合、ステップ S 5 4 において、アービトレーションコントローラ 7 は、コンパレータ 4 - 1 から端子 7 b に True の信号が入力されているか否かを判定し、例えば、True の信号が入力されていると判定した場合、その処理は、ステップ S 5 5 に進む。

【 0 0 8 3 】

ステップ S 5 5 において、アービトレーションコントローラ 7 は、コンパレータ 4 - 2 から端子 7 c に True の信号が入力されているか否かを判定し、例えば、True の信号が入力されていると判定した場合、その処理は、ステップ S 5 6 に進む。

【 0 0 8 4 】

ステップ S 5 6 において、アービトレーションコントローラ 7 は、コンパレータ 6 - 1 から端子 7 h に True の信号が入力されているか否かを判定し、例えば、True の信号が入力されていると判定した場合、その処理は、ステップ S 5 7 に進む。

【 0 0 8 5 】

ステップ S 5 7 において、アービトレーションコントローラ 7 は、コンパレータ 6 - 2 から端子 7 i に True の信号が入力されているか否かを判定し、例えば、True の信号が入力されていると判定した場合、その処理は、ステップ S 5 8 に進む。

【 0 0 8 6 】

ステップ S 5 8 において、アービトレーションコントローラ 7 は、ラウンドロビン方式によりメモリ 7 a に記憶されている過去の記録から、最近バス 8 が取得されていないモジュールを優先して優先順位を設定する。すなわち、ステップ S 5 4 乃至 S 5 8 の処理により、符号化部 1、および、復号部 2 は、それぞれ所定の値 C 2 に対応する時間以上にリクエスト信号を出力してから確認信号を待ち続けていることになるので、いずれも待ち時間は同一であると判断し、待ち時間での比較をせずに、過去のバス 8 の取得状況に応じて優先順位を設定する。

【 0 0 8 7 】

すなわち、今の場合、モジュールは、符号化部 1 と復号部 2 の 2 個であるので、直前で、符号化部 1 にバス 8 を取得させた場合、復号部 2 の優先順位が優先され、優先順位の第 1 位が復号部 2 に設定され、第 2 位に符号化部 1 が設定される。また、逆に、直前に復号部 2 にバス 8 を取得させた場合、符号化部 1 の優先順位が優先的に設定され、優先順位の第 1 位に符号化部 1 が設定され、第 2 位に復号部 2 が設定される。

【 0 0 8 8 】

ステップ S 5 9 において、アービトレーションコントローラ 7 は、設定された優先順位の情報に基づいて、最も優先順位の高いモジュール、すなわち、優先順位が第 1 位のモジュールに対して確認信号を送信し、バス 8 の取得させる。すなわち、図 2 の場合、最も優先順位の高いモジュールが符号化部 1 であった場合、端子 7 d から符号化部 1 に対して確認信号を出力する。また、逆に、最も優先順位の高いモジュールが復号部 2 であった場合、端子 7 f から復号部 2 に対して確認信号を出力する。

【 0 0 8 9 】

ステップ S 5 1 において、符号化部 1 からのみリクエスト信号が入力されていると判定された場合、ステップ S 6 0 において、符号化部 1 に優先順位が優先して設定される。すなわち、ステップ S 5 1 において、符号化部 1 からのみリクエスト信号が受信されているということは、調停の必要が無く、符号化部 1 に優先的に優先順位が設定される。

【 0 0 9 0 】

ステップ S 5 2 において、復号部 2 からのみリクエスト信号が入力されていると判定された場合、ステップ S 6 1 において、復号部 1 に優先順位が優先して設定される。すなわち、ステップ S 5 2 において、復号部 2 からのみリクエスト信号が受信されているということは、調停の必要が無く、復号部 2 に優先的に優先順位が設定される。

【 0 0 9 1 】

ステップ S 5 4 において、コンパレータ 4 - 1 から端子 7 b に入力される信号が True ではないと判定された場合、ステップ S 6 2 において、アービトレーションコントローラ 7 は、コンパレータ 6 - 1 から端子 7 g に入力される信号が True であるか否かを判定する。例えば、コンパレータ 6 - 1 から端子 7 g に入力される信号が True ではないと判定された場合、その処理は、ステップ S 5 8 に進み、ラウンドロビン方式で優先順位が設定される。

【 0 0 9 2 】

すなわち、ステップ S 5 4, S 6 2 において、いずれも判定が No となるということは、符号化部 1、および、復号部 2 のいずれもリクエスト信号を出力してから、所定の値 C 1 に対応する待ち時間を経過しておらず、待ち時間が同一であると判定されるため、待ち時間による比較をせずに、ラウンドロビン方式で優先順位が設定される。

【 0 0 9 3 】

ステップ S 6 2 において、コンパレータ 6 - 1 から端子 7 g に入力される信号が True であると判定された場合、その処理は、ステップ S 6 1 に進み、復号部 2 に優先的に優先順位が設定される。すなわち、ステップ S 5 4 において、No で、ステップ S 6 2 において、Yes であるということは、符号化部 1 では、リク

エスト信号を送信してから所定の値 C 1 に対応する待ち時間が発生していないことになるが、復号部 2 では、リクエスト信号を送信してから所定の値 C 1 に対応する待ち時間が発生していることになるため、復号部 2 の方が、符号化部 1 よりもリクエスト信号を送信してからの待ち時間が長いことになるので、復号部 2 に対して優先的に優先順位が設定される。

【 0 0 9 4 】

ステップ S 5 5 において、コンパレータ 4 - 2 から端子 7 c に入力される信号が True ではないと判定された場合、ステップ S 6 3 において、アービトレーションコントローラ 7 は、コンパレータ 6 - 1 から端子 7 h に入力される信号が True であるか否かを判定し、コンパレータ 6 - 1 から端子 7 h に入力される信号が True ではないと判定した場合、その処理は、ステップ S 6 0 に進む。すなわち、ステップ S 5 6 において、No で、かつ、ステップ S 6 3 において、No であるということは、符号化部 1 は、所定の値 C 1 に対応する時間以上の待ち時間が発生していることになるが、復号部 2 は、所定の値 C 1 に対応する時間にも満たない時間しか待ち時間が発生しないことになり、符号化部 1 の方がリクエスト信号を出力してからより長い待ち時間が発生しているので、ステップ S 6 0 において、符号化部 1 に対して優先的に優先順位が設定される。

【 0 0 9 5 】

ステップ S 6 3 において、コンパレータ 6 - 1 から端子 7 h に入力される信号が True であると判定した場合、ステップ S 6 4 において、アービトレーションコントローラ 7 は、コンパレータ 6 - 2 から端子 7 i に入力される信号が True であるか否かを判定し、コンパレータ 6 - 2 から端子 7 i に入力される信号が True ではないと判定した場合、その処理は、ステップ S 5 8 に進む。すなわち、ステップ S 5 5 において、No で、かつ、ステップ S 6 4 において、No であるということは、符号化部 1、および、復号部 2 のいずれも、リクエスト信号を送信してから、所定の値 C 1 に対応する時間までの待ち時間が発生し、所定の値 C 2 に対応する時間までの待ち時間は発生していないことになるため、発生している待ち時間は同一であると判定し、待ち時間による比較を行わずに、ステップ S 5 8 において、ラウンドロビン方式により優先順位が設定される。

【 0 0 9 6 】

ステップ S 5 6、または、S 5 7において、コンパレータ 6 - 1 から端子 7 h に True の信号が入力されていない、または、コンパレータ 6 - 2 から端子 7 i に True の信号が入力されていないと判定された場合、その処理は、ステップ S 6 0 に進み、符号化部 1 に優先的に優先順位が設定される。すなわち、ステップ S 5 6 または S 5 7 が N o であるということは、符号化部 1 は、所定の値 C 2 に対応する以上の待ち時間が発生しているにもかかわらず、符号化部 2 は、所定の値 C 1 に対応する時間にも満たない待ち時間である（ステップ S 5 6 が N o の場合）か、または、所定の値 C 2 に対応する時間にも満たない待ち時間しか発生していないことになるので、いずれにおいても、待ち時間は、復号部 2 よりも符号化部 1 の方が長いことになるので、符号化部 1 に優先的に優先順位を設定する。

【 0 0 9 7 】

以上の処理により、複数の複数のモジュールが存在する場合でも、リクエスト信号を送信してからの待ち時間に応じて、効率よく優先順位を設定することができるので、例えば、図 1 で示したようなタイミングで符号化部 1 と復号部 2 の処理が実行されるようなとき、処理効率を向上させることができる。

【 0 0 9 8 】

図 8 A、B は、上述の処理により優先順位を設定して、バス 8 の取得をモジュール毎に調整したときのリクエスト信号の送信タイミングと各モジュールの処理状態を示すタイミングチャートである。

【 0 0 9 9 】

尚、図 8 A、B においては、説明の都合上、図 8 A が符号化部 1 の処理を示し、図 8 B が復号部 2 の処理を示すタイミングチャートであるものとするが、入れ替わっても同様である。また、図 8 A、B においては、実線で示す処理のタイミングは、ラウンドロビン方式でバスアービトレーションを行った場合のタイミングチャートであり、図 1 で示したタイミングチャートと同様のものであり、ここでは、第 1 モジュールが符号化部 1 を示し、第 2 モジュールが復号部 2 に相当する。また、点線で示した処理のタイミングは、図 7 のフローチャートを参照して説明した方法によりバスアービトレーション処理（バス取得の調停処理）を行っ

た場合のタイミングチャートである。さらに、点線で示す図 7 のフローチャートに準じた方法で実行されたバスアービトレーションによりなされた処理やリクエスト信号を送信するタイミングには、「'」を付している。

【0 1 0 0】

すなわち、時刻 t_1 でリクエスト信号 R_1' が送信されると、このときは、符号化部 1 からのリクエストだけであるので（図 7 のステップ S 5 1 が Y e s の場合）、符号化部 1 に確認信号が送信されて、処理 1' が実行され、その処理が終了する時刻 t_2 のタイミングで、新たにリクエスト信号 R_2' が送信されると、このときも、符号化部 1 からのリクエストだけであるので（図 7 のステップ S 5 1 が Y e s の場合）、符号化部 1 に確認信号が送信されて、処理 2' が実行される。さらに、処理 2' が終了する時刻 t_3 において、符号化部 1 からリクエスト信号 R_3' が送信されると同時に、復号部 2 がリクエスト信号 R_{11}' を送信する。このとき、いずれも所定の値 C_1 に対応する時間だけリクエスト信号が送信されてから経過していないので（ステップ S 5 4 が N o、でかつ、ステップ S 6 2 が N o）、ラウンドロビン方式で優先順位が設定されて、復号部 2 に確認信号が送信されて（直前では符号化部 1 が優先されるように優先順位が設定されているため）、処理 1 1' が実行される。処理 1 1' が終了する時刻 t_4 においては、リクエスト信号を送信しているのは、符号化部 1 だけとなるので（ステップ S 5 1 が Y e s）、符号化部 1 に確認信号を送信し、処理 3' が実行される。

【0 1 0 1】

処理 3' が実行された後のタイミングである時刻 t_5 において、符号化部 1 がリクエスト信号 R_4' を出力すると、そのタイミングでリクエスト信号を出力しているモジュールは、符号化部 1 のみであるので（ステップ S 5 1 が Y e s）、符号化部 1 に確認信号が送信されて、処理 4' が実行される。

【0 1 0 2】

処理 4' が終了する時刻 t_6 において、復号部 2 は、リクエスト信号 R_{12}' を出力する。このとき、リクエスト信号を送信しているモジュールは、復号部 2 だけである（ステップ S 5 2 が Y e s）ので、復号部 2 に確認信号が送信されて、処理 1 2' が実行される。

【 0 1 0 3 】

処理 4' が終了する時刻 t 6 から 2 タイミング（時刻 t 6 と時刻 t 8 との時間差）遅れた時刻 t 8 において、符号化部 1 は、リクエスト信号 R 5' を出力する（図 1 においても、処理 4 の後、2 タイミング遅れてリクエスト信号 R 5 が出力されている）。このとき、リクエスト信号を送信しているモジュールは、符号化部 1 だけである（ステップ S 5 1 が Y e s）ので、符号化部 1 に確認信号が送信されて、処理 5' が実行される。

【 0 1 0 4 】

処理 5' が終了する時刻 t 9 のタイミングで、復号部 2 は、リクエスト信号 R 1 3' を出力する。このとき、リクエスト信号を送信しているモジュールは、復号部 2 だけである（ステップ S 5 2 が Y e s）ので、復号部 2 に確認信号が送信されて、処理 1 2' が実行される。

【 0 1 0 5 】

さらに、処理 1 3' が終了する時刻 t 1 0 において、復号部 2 はリクエスト信号 R 1 4' を出力する。このときも、リクエスト信号を送信しているモジュールは、復号部 2 だけである（ステップ S 5 2 が Y e s）ので、復号部 2 に確認信号が送信されて、処理 1 4' が実行される。

【 0 1 0 6 】

処理 1 4' が終了する時刻 t 1 1 において、復号部 2 は、リクエスト信号 R 1 5' を出力する。同時に、符号化部 1 も、リクエスト信号 R 6' を出力する。このとき、いずれもリクエスト信号が送信されてから所定の値 C 1 に対応する時間だけ経過していないので（ステップ S 5 4 が N o、でかつ、ステップ S 6 2 が N o）、ラウンドロビン方式で優先順位が設定されて、符号化部 1 に確認信号が送信されて（直前では復号部 2 が優先されるように優先順位が設定されて、処理 1 4' が実行されているため）、処理 6' が実行される。処理 6' が終了する時刻 t 1 2 においては、リクエスト信号を送信しているのは、復号部 2 だけとなるので（ステップ S 5 2 が Y e s）、復号部 2 に確認信号が送信され、処理 1 5' が実行される。さらに、処理 1 5' が終了する時刻 t 1 3 において、復号部 2 はリクエスト信号 R 1 6' を出力するが、このときも、リクエスト信号を出力してい

るモジュールは、復号部 2 のみであるので、復号部 2 に確認信号が送信されて、処理 1 6' が実行されて処理が終了する。

【 0 1 0 7 】

結果として、ラウンドロビン方式で優先順位を設定するようにしたときには、図 1 A, B の双方の処理を同時に実行すると、その処理時間は、時間 t 1 乃至 t 1 8 であったのに対して、図 7 のフローチャートを参照して説明した処理によると、処理時間は、時間 t 1 乃至 t 1 4 となる。

【 0 1 0 8 】

このようにバスアービトレーションを効率よく行うようにすることができるので、LSI 全体として処理時間を短縮することができ、結果的に、処理速度を向上させるようにすることができる。

【 0 1 0 9 】

また、図 7 のフローチャートを参照して説明した優先順位の設定方法は、リクエスト信号が複数のモジュールから入力されていた場合、コンパレータ 4 - 1, 4 - 2, 6 - 1, 6 - 2 から端子 7 b, 7 c, 7 h, 7 i に入力される信号のパターンにより分けられることになる。

【 0 1 1 0 】

図 9 は、これらのパターン毎に設定される優先順位をまとめたものである。尚、図 9 においては、True の信号が「1」、False の信号が「0」としてそれぞれ示されている。また、符号化部 1 をモジュール M 0、復号部 2 をモジュール M 1 として示している。さらに。図中左列は、優先順位の設定状態を示しており、左から優先順位が第 1 位、第 2 位を示しており、R (M 0, M 1) は、モジュール M 0, M 1 の優先順位をラウンドロビン方式で設定することを示している。

【 0 1 1 1 】

図 9 においては、コンパレータ (Comp) 4 - 1, 4 - 2, 6 - 1, 6 - 2 の出力信号が、(0, 0, 0, 0), (1, 0, 1, 0), (1, 1, 1, 1) の場合、ラウンドロビン方式で優先順位が設定され、(1, 0, 0, 0), (1, 1, 0, 0), (1, 1, 1, 0) の場合、モジュール M 0 が優先順位第 1 位に設定され、モジュール M 1 が第 2 位に設定され、(1, 0, 1, 1), (0, 0,

1, 1), (0, 0, 1, 0) の場合、モジュール M 1 が優先順位第 1 位に設定され、モジュール M 0 が第 2 位に設定されることが示されている。

【0 1 1 2】

尚、上述の (* 1, * 2, * 3, * 4) は、* 1 がコンパレータ 4 - 1 から出力される信号を、* 2 がコンパレータ 4 - 2 から出力される信号を、* 3 がコンパレータ 6 - 1 から出力される信号を、* 4 がコンパレータ 6 - 2 から出力される信号をそれぞれ示している。

【0 1 1 3】

従って、アービトレーションコントローラ 7 は、これらのパターンをテーブルとして予め記憶しておき、このパターンに応じて確認信号を各モジュール、すなわち、今の場合、符号化部 1、または、復号部 2 に送信するようにしてもよい。

【0 1 1 4】

また、以上の例においては、モジュールとして、符号化部 1、および、復号部 2 の 2 個から構成される LSI の構成例について説明してきたが、モジュールの数は、それ以上の数であってもよく、例えば、図 1 0 で示すように、モジュールが 3 個の LSI の構成であってもよい。

【0 1 1 5】

図 1 0 において、モジュール 1 0 1 乃至 1 0 3 は、図 2 の符号化部 1、または、復号部 2 に対応するものであり、以下においては、モジュール M 0 乃至 M 2 とも称する。また、カウンタ 1 0 4 乃至 1 0 6 は、図 2 のカウンタ 3 または 5 と対応するものである。さらに、コンパレータ 1 0 7 - 1 乃至 1 0 7 - 3, 1 0 8 - 1 乃至 1 0 8 - 3, 1 0 9 - 1 乃至 1 0 9 - 3 は、図 2 のコンパレータ 4 - 1, 4 - 2, 6 - 1、または 6 - 2 と対応するものである。ただし、コンパレータ 1 0 7 - 1 乃至 1 0 7 - 3, 1 0 8 - 1 乃至 1 0 8 - 3, 1 0 9 - 1 乃至 1 0 9 - 3 は、コンパレータ 1 0 7 - 1 乃至 1 0 9 - 1 が所定の値 C 1 とカウント値を比較し、コンパレータ 1 0 7 - 2 乃至 1 0 9 - 2 が所定の値 C 2 とカウント値を比較し、コンパレータ 1 0 7 - 3 乃至 1 0 9 - 3 が所定の値 C 3 とカウント値を比較しており、ここでは、 $C 1 < C 2 < C 3$ であるものとする。また、アービトレーションコントローラ 1 1 0 は、図 2 のアービトレーションコントローラ 7 に対

応するものであり、バス 1 1 1 は、バス 8 に対応するものであり、外部メモリ 1 1 2 は、外部メモリ 9 に対応するものである。

【 0 1 1 6 】

このような構成の場合、コンパレータ 1 0 7 - 1 乃至 1 0 7 - 3, 1 0 8 - 1 乃至 1 0 8 - 3, 1 0 9 - 1 乃至 1 0 9 - 3 から出力される信号に対応して、図 1 1 乃至図 1 3 で示されるようにアービトレーションコントローラ 1 1 1 が優先順位を設定し、バス 1 1 1 を取得させて、外部メモリ 1 1 2 を適宜利用して各種の処理を実行させる。

【 0 1 1 7 】

すなわち、図 1 1 で示すように、コンパレータ 1 0 7 - 1 乃至 1 0 7 - 3, 1 0 8 - 1 乃至 1 0 8 - 3, 1 0 9 - 1 乃至 1 0 9 - 3 からの出力信号が、(0, 0, 0, 0, 0, 0, 0, 0), (1, 0, 0, 1, 0, 0, 1, 0, 0), (1, 1, 0, 1, 1, 0, 1, 1, 0), (1, 1, 1, 1, 1, 1, 1, 1, 1) の場合、モジュール M 0 乃至 M 2 はラウンドロビン方式により優先順位が設定される。

【 0 1 1 8 】

また、コンパレータ 1 0 7 - 1 乃至 1 0 7 - 3, 1 0 8 - 1 乃至 1 0 8 - 3, 1 0 9 - 1 乃至 1 0 9 - 3 からの出力信号が、(1, 0, 0, 0, 0, 0, 0, 0, 0), (1, 1, 0, 0, 0, 0, 0, 0, 0), (1, 1, 0, 1, 0, 0, 1, 0, 0), (1, 1, 1, 0, 0, 0, 0, 0, 0), (1, 1, 1, 1, 0, 0, 1, 0, 0), (1, 1, 1, 1, 1, 0, 1, 1, 0) の場合、第 1 位が、モジュール M 0 であり、第 2 位、および、第 3 位は、モジュール M 1, M 2 からラウンドロビン方式により設定される。

【 0 1 1 9 】

さらに、コンパレータ 1 0 7 - 1 乃至 1 0 7 - 3, 1 0 8 - 1 乃至 1 0 8 - 3, 1 0 9 - 1 乃至 1 0 9 - 3 からの出力信号が、(0, 0, 0, 1, 0, 0, 1, 0, 0), (0, 0, 0, 1, 1, 0, 1, 1, 0), (0, 0, 0, 1, 1, 1, 1, 1, 1), (1, 0, 0, 1, 1, 0, 1, 1, 0), (1, 0, 0, 1, 1, 1, 1, 1, 1), (1, 1, 0, 1, 1, 1, 1, 1, 1) の場

合、第1位、および、第2位は、モジュールM1, M2からラウンドロビン方式により設定され、第3位が、モジュールM0として設定される。

【0 1 2 0】

また、コンパレータ1 0 7 - 1乃至1 0 7 - 3, 1 0 8 - 1乃至1 0 8 - 3, 1 0 9 - 1乃至1 0 9 - 3からの出力信号が、(1, 1, 0, 1, 0, 0, 0, 0, 0), (1, 1, 1, 1, 1, 0, 1, 0, 0), (1, 1, 1, 1, 1, 0, 1, 1, 0), (1, 1, 1, 1, 1, 0, 0, 0, 0)の場合、第1位はモジュールM0、第2位は、モジュールM1、第3位はモジュールM2として設定される。

【0 1 2 1】

さらに、コンパレータ1 0 7 - 1乃至1 0 7 - 3, 1 0 8 - 1乃至1 0 8 - 3, 1 0 9 - 1乃至1 0 9 - 3からの出力信号が、(1, 1, 0, 0, 0, 0, 1, 0, 0), (1, 1, 1, 0, 0, 0, 1, 0, 0), (1, 1, 1, 0, 0, 0, 1, 1, 0), (1, 1, 1, 1, 0, 0, 1, 1, 0)の場合、第1位はモジュールM0、第2位は、モジュールM2、第3位はモジュールM1として設定される。

【0 1 2 2】

また、図12で示すように、コンパレータ1 0 7 - 1乃至1 0 7 - 3, 1 0 8 - 1乃至1 0 8 - 3, 1 0 9 - 1乃至1 0 9 - 3からの出力信号が、(0, 0, 0, 1, 0, 0, 0, 0, 0), (0, 0, 0, 1, 1, 0, 0, 0, 0), (1, 0, 0, 1, 1, 0, 1, 0, 0), (0, 0, 0, 1, 1, 1, 0, 0, 0), (1, 0, 0, 1, 1, 1, 1, 0, 0), (1, 1, 0, 1, 1, 1, 1, 1, 0)の場合、第1位が、モジュールM1であり、第2位、および、第3位は、モジュールM0, M2からラウンドロビン方式により設定される。

【0 1 2 3】

さらに、コンパレータ1 0 7 - 1乃至1 0 7 - 3, 1 0 8 - 1乃至1 0 8 - 3, 1 0 9 - 1乃至1 0 9 - 3からの出力信号が、(1, 0, 0, 0, 0, 0, 1, 0, 0), (1, 1, 0, 0, 0, 0, 1, 1, 0), (1, 1, 1, 0, 0, 0, 1, 1, 1), (1, 1, 0, 1, 0, 0, 1, 1, 0), (1, 1, 1, 0, 0, 0, 1, 1, 0)

, 1, 0, 0, 1, 1, 1), (1, 1, 1, 1, 1, 0, 1, 1, 1) の場合、第 1 位、および、第 2 位は、モジュール M 0, M 2 からラウンドロビン方式により設定され、第 3 位が、モジュール M 1 として設定される。

【 0 1 2 4 】

また、コンパレータ 1 0 7 - 1 乃至 1 0 7 - 3, 1 0 8 - 1 乃至 1 0 8 - 3, 1 0 9 - 1 乃至 1 0 9 - 3 からの出力信号が、(1, 0, 0, 1, 1, 0, 0, 0, 0), (1, 0, 0, 1, 1, 1, 0, 0, 0), (1, 1, 0, 1, 1, 1, 0, 0, 0), (1, 1, 0, 1, 1, 1, 1, 0, 0) の場合、第 1 位はモジュール M 1、第 2 位は、モジュール M 0、第 3 位はモジュール M 2 として設定される。

【 0 1 2 5 】

さらに、コンパレータ 1 0 7 - 1 乃至 1 0 7 - 3, 1 0 8 - 1 乃至 1 0 8 - 3, 1 0 9 - 1 乃至 1 0 9 - 3 からの出力信号が、(0, 0, 0, 1, 1, 0, 1, 0, 0), (0, 0, 0, 1, 1, 1, 1, 0, 0), (0, 0, 0, 1, 1, 1, 1, 1, 0), (1, 0, 0, 1, 1, 1, 1, 1, 0) の場合、第 1 位はモジュール M 1、第 2 位は、モジュール M 2、第 3 位はモジュール M 0 として設定される。

【 0 1 2 6 】

また、図 1 3 で示すように、コンパレータ 1 0 7 - 1 乃至 1 0 7 - 3, 1 0 8 - 1 乃至 1 0 8 - 3, 1 0 9 - 1 乃至 1 0 9 - 3 からの出力信号が、(0, 0, 0, 0, 0, 1, 0, 0), (0, 0, 0, 0, 0, 0, 1, 1, 0), (1, 0, 0, 1, 0, 0, 1, 1, 0), (0, 0, 0, 0, 0, 0, 1, 1, 1), (1, 0, 0, 1, 0, 0, 1, 1, 1), (1, 1, 0, 1, 1, 0, 1, 1, 1) の場合、第 1 位が、モジュール M 2 であり、第 2 位、および、第 3 位は、モジュール M 0, M 1 からラウンドロビン方式により設定される。

【 0 1 2 7 】

さらに、コンパレータ 1 0 7 - 1 乃至 1 0 7 - 3, 1 0 8 - 1 乃至 1 0 8 - 3, 1 0 9 - 1 乃至 1 0 9 - 3 からの出力信号が、(1, 0, 0, 1, 0, 0, 0, 0, 0), (1, 1, 0, 1, 1, 0, 0, 0, 0), (1, 1, 1, 1, 1, 0, 0, 0, 0) の場合、第 1 位は、モジュール M 2 であり、第 2 位、および、第 3 位は、モジュール M 0, M 1 からラウンドロビン方式により設定される。

, 1, 0, 0, 0), (1, 1, 0, 1, 1, 0, 1, 0, 0), (1, 1, 1, 1, 1, 1, 1, 0, 0), (1, 1, 1, 1, 1, 1, 1, 1, 0) の場合、第 1 位、および、第 2 位は、モジュール M 0, M 1 からラウンドロビン方式により設定され、第 3 位が、モジュール M 2 として設定される。

【 0 1 2 8 】

また、コンパレータ 1 0 7 - 1 乃至 1 0 7 - 3, 1 0 8 - 1 乃至 1 0 8 - 3, 1 0 9 - 1 乃至 1 0 9 - 3 からの出力信号が、(1, 0, 0, 0, 0, 0, 1, 1, 0), (1, 0, 0, 0, 0, 0, 1, 1, 1), (1, 1, 0, 0, 0, 0, 1, 1, 1), (1, 1, 0, 1, 0, 0, 1, 1, 1) の場合、第 1 位はモジュール M 2、第 2 位は、モジュール M 0、第 3 位はモジュール M 1 として設定される。

【 0 1 2 9 】

さらに、コンパレータ 1 0 7 - 1 乃至 1 0 7 - 3, 1 0 8 - 1 乃至 1 0 8 - 3, 1 0 9 - 1 乃至 1 0 9 - 3 からの出力信号が、(0, 0, 0, 1, 0, 0, 1, 1, 0), (0, 0, 0, 1, 0, 0, 1, 1, 1), (1, 0, 0, 1, 1, 0, 1, 1, 1), (0, 0, 0, 1, 1, 0, 1, 1, 1) の場合、第 1 位はモジュール M 2、第 2 位は、モジュール M 1、第 3 位はモジュール M 0 として設定される。

【 0 1 3 0 】

このような処理により、図 1 0 においては、リクエスト信号を送信してから確認信号が送信されてくるまでの待ち時間が、所定の値 C 1 に対する待ち時間 S、所定の値 C 2 に対応する待ち時間 M、所定の値 C 3 に対応する待ち時間 L が設定され、 $C 1 < C 2 < C 3$ であることから $S < M < L$ となるので、各モジュール M 0 乃至 M 2 毎の待ち時間の組み合わせに応じて、モジュール M 0 乃至 M 2 の優先順位が設定されて、その中で第 1 位に設定されているモジュールに対してバス 1 1 1 が取得されるようになり、外部メモリ 1 1 2 に対してデータを記録させると共に読み出せるように調停を行うことができるので、L S I 内の各モジュールの処理を全体として効率よく実行させることが可能となり、結果として、処理速度を向上させることが可能となる。

【 0 1 3 1 】

尚、以上の例においては、符号化部 1 と復号部 2 をモジュールとして設定した場合の LSI の構成例について説明してきたが、モジュールはこれ以外のものであってもよい。また、以上においては、LSI における各モジュールを例として説明してきたが、LSI 上のモジュールに限らず、例えば、プログラム上の処理を実行する際に、CPU により処理の優先順位を設定する場合に応用してもよい。

【 0 1 3 2 】

さらに、図 2 の構成例における、画像入力部 2 1、動き予測部 2 2、符号化処理部 2 3、ビットストリーム出力部 2 4、ビットストリーム入力部 4 2、復号処理部 4 3、動き補償部 4 4、および、画像出力部 4 5 のそれぞれをモジュールとして構成するようにしてもよい。

【 0 1 3 3 】

以上によれば、リクエスト信号を送信してから確認信号が送信されてくるまでの待ち時間の長さに応じて、モジュールが取得するバスの優先順位が設定されるので、適正な優先順位を設定することができ、効率のよいバスアービトレーション（バスの取得を調停する処理）を実現させることが可能となる。

【 0 1 3 4 】

上述した一連の処理は、ハードウェアにより実行させることもできるが、ソフトウェアにより実行させることもできる。一連の処理をソフトウェアにより実行させる場合には、そのソフトウェアを構成するプログラムが、専用のハードウェアに組み込まれているコンピュータ、または、各種のプログラムをインストールすることで、各種の機能を実行させることが可能な、例えば汎用のパーソナルコンピュータなどに記録媒体からインストールされる。

【 0 1 3 5 】

図 1 2 は、図 2 で示した LSI を、それぞれソフトウェアにより実現する場合のパーソナルコンピュータの一実施の形態の構成を示している。パーソナルコンピュータの CPU 2 0 1 は、パーソナルコンピュータの動作の全体を制御する。また、CPU 2 0 1 は、バス 2 0 4 および入出力インタフェース 2 0 5 を介してユーザからキーボードやマウスなどからなる入力部 2 0 6 から指令が入力されると、そ

れに対応してROM(Read Only Memory) 2 0 2 に格納されているプログラムを実行する。あるいはまた、CPU 2 0 1 は、ドライブ 2 1 0 に接続された磁気ディスク 2 2 1、光ディスク 2 2 2、光磁気ディスク 2 2 3、または半導体メモリ 2 2 4 から読み出され、記憶部 2 0 8 にインストールされたプログラムを、RAM(Random Access Memory) 2 0 3 にロードして実行し、出力部 2 0 7 が実行結果を出力する。さらに、CPU 2 0 1 は、通信部 2 0 9 を制御して、外部と通信し、データの授受を実行する。

【 0 1 3 6 】

プログラムが記録されている記録媒体は、図 1 2 に示すように、コンピュータとは別に、ユーザにプログラムを提供するために配布される、プログラムが記録されている磁気ディスク 2 2 1 (フレキシブルディスクを含む)、光ディスク 2 2 2 (CD-ROM(Compact Disc-Read Only Memory), DVD (Digital Versatile Disc)を含む)、光磁気ディスク 2 2 3 (MD (Mini-Disc)を含む)、もしくは半導体メモリ 2 3 4 などよりなるパッケージメディアにより構成されるだけでなく、コンピュータに予め組み込まれた状態でユーザに提供される、プログラムが記録されているROM 2 0 2 や、記憶部 2 0 8 に含まれるハードディスクなどで構成される。

【 0 1 3 7 】

尚、本明細書において、記録媒体に記録されるプログラムを記述するステップは、記載された順序に沿って時系列的に行われる処理は、もちろん、必ずしも時系列的に処理されなくとも、並列的あるいは個別に実行される処理を含むものである。

【 0 1 3 8 】

【発明の効果】

本発明によれば、適正な優先順位を設定することができ、効率のよいバスアービトレーション (バスの取得を調停する処理) を実現させることが可能となる。

【図面の簡単な説明】

【図 1】

従来のバスアービトレーションによる複数のモジュールによる処理のタイミン

グを示すタイミングチャートである。

【図 2】

本発明を適用したLSIの構成を示すブロック図である。

【図 3】

符号化制御処理を説明するフローチャートである。

【図 4】

カウント処理を説明するフローチャートである。

【図 5】

比較処理を説明するフローチャートである。

【図 6】

復号制御処理を説明するフローチャートである。

【図 7】

調停処理を説明するフローチャートである。

【図 8】

本発明を適用したバスアービトレーションによる複数のモジュールによる処理のタイミングを示すタイミングチャートである。

【図 9】

図 2 のコンパレータの出力信号のパターンと優先順位の関係を示す図である。

【図 1 0】

モジュールが3個の場合のLSIの構成例を示すブロック図である。

【図 1 1】

図 1 0 のコンパレータの出力信号のパターンと優先順位の関係を示す図である。
。

【図 1 2】

図 1 0 のコンパレータの出力信号のパターンと優先順位の関係を示す図である。
。

【図 1 3】

図 1 0 のコンパレータの出力信号のパターンと優先順位の関係を示す図である。
。

【図 1 4】

媒体を説明する図である。

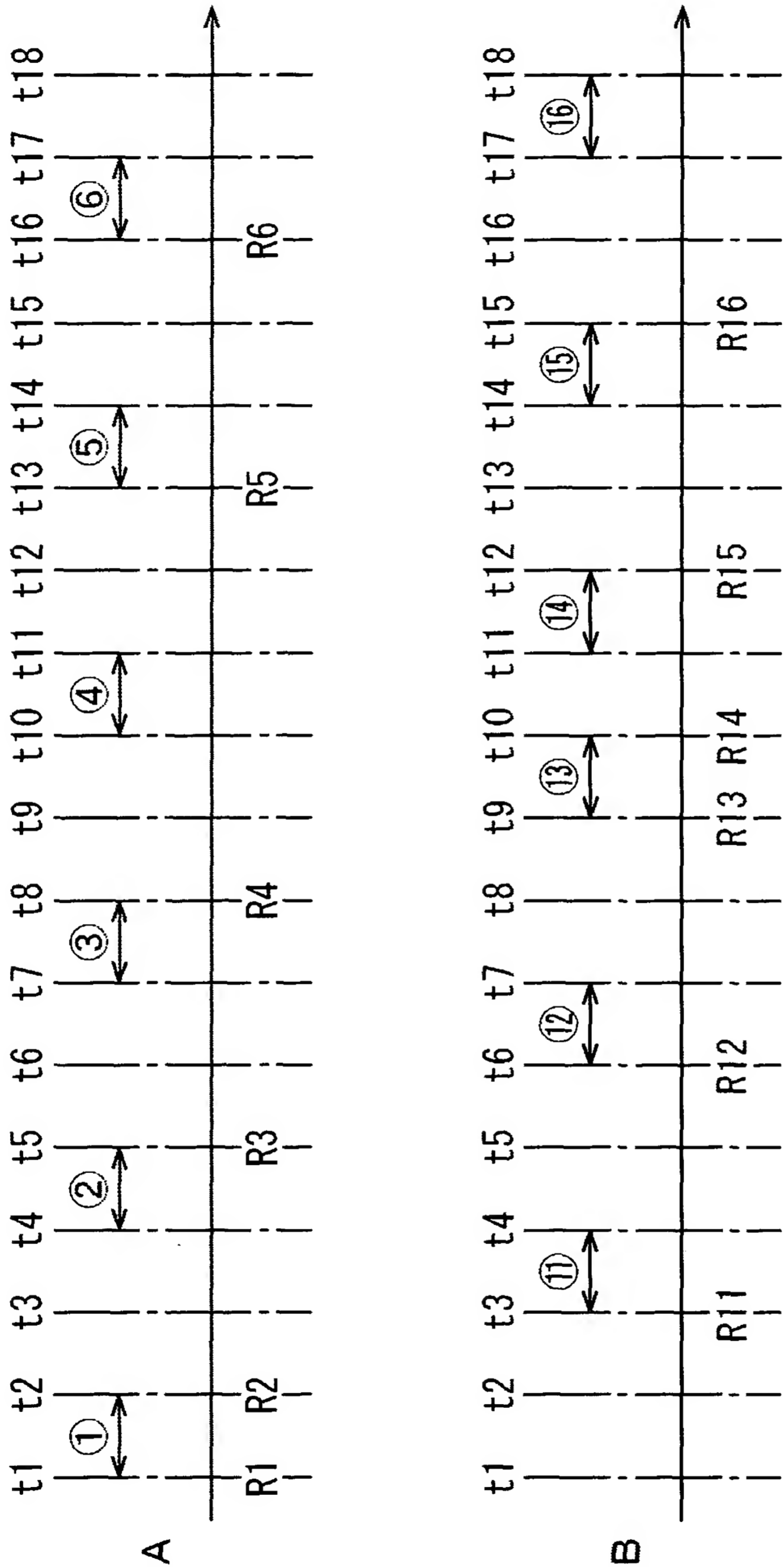
【符号の説明】

1 符号化部, 2 復号部, 3 カウンタ, 3 a, 3 b 端子, 4-1, 4-2 コンパレータ, 5 カウンタ, 5 a, 5 b 端子, 6-1, 6-2 コンパレータ, 7 アービトレーションコントローラ, 7 a メモリ, 7 b 乃至 7 i 端子, 8 バス, 9 外部メモリ, 21 画像入力部, 22 動き予測部, 23 符号化処理部, 24 ビットストリーム出力部, 25 符号化制御部, 25 a, 25 b 端子, 41 復号制御部, 42 ビットストリーム入力部, 43 復号処理部, 44 動き補償部, 45 画像出力部, 101 乃至 103 モジュール, 104 乃至 106 カウンタ, 107-1 乃至 107-3, 108-1 乃至 108-3, 109-1 乃至 109-3 コンパレータ, 110 アービトレーションコントローラ, 111 バス, 112 外部メモリ

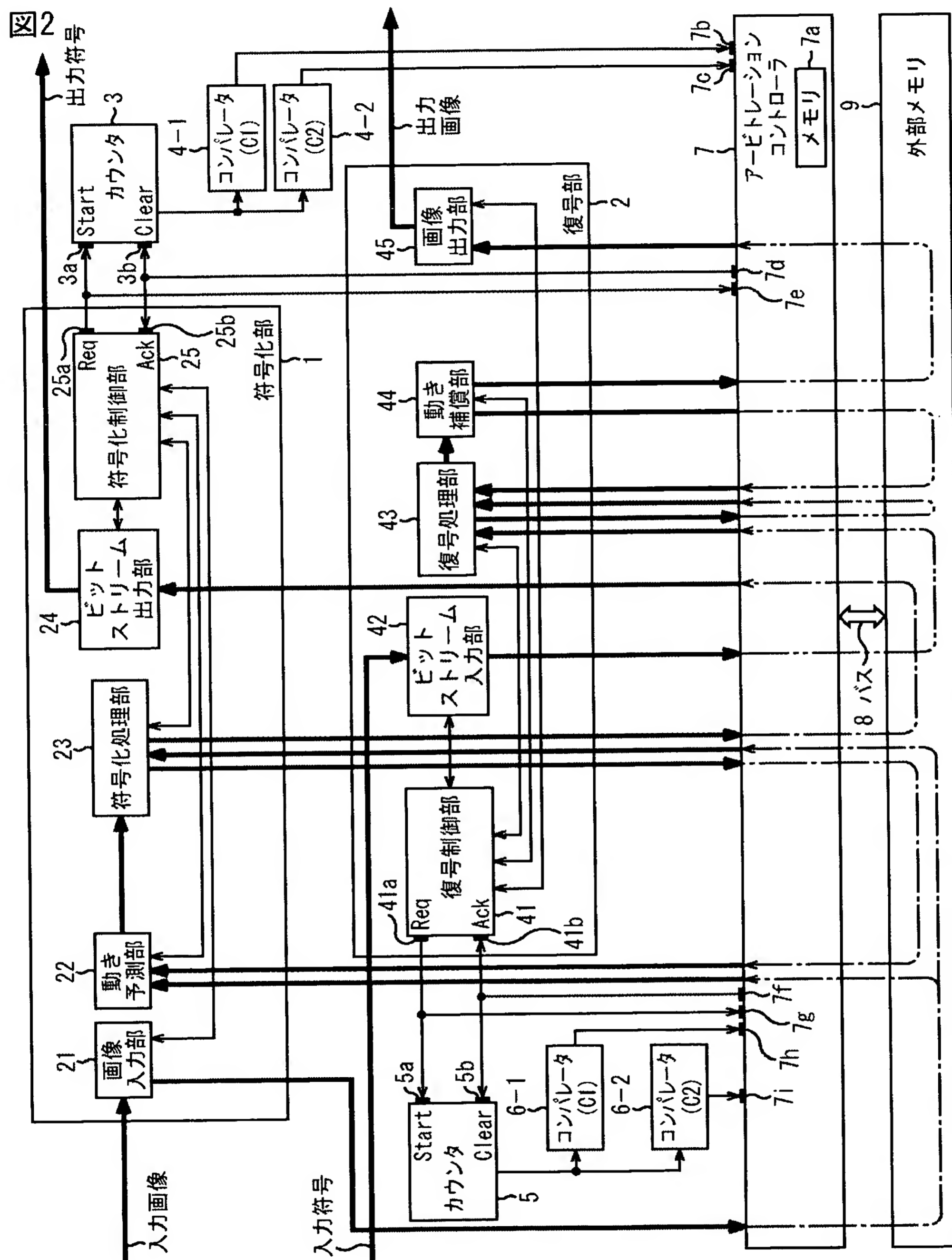
【書類名】 図面

【図 1】

図1

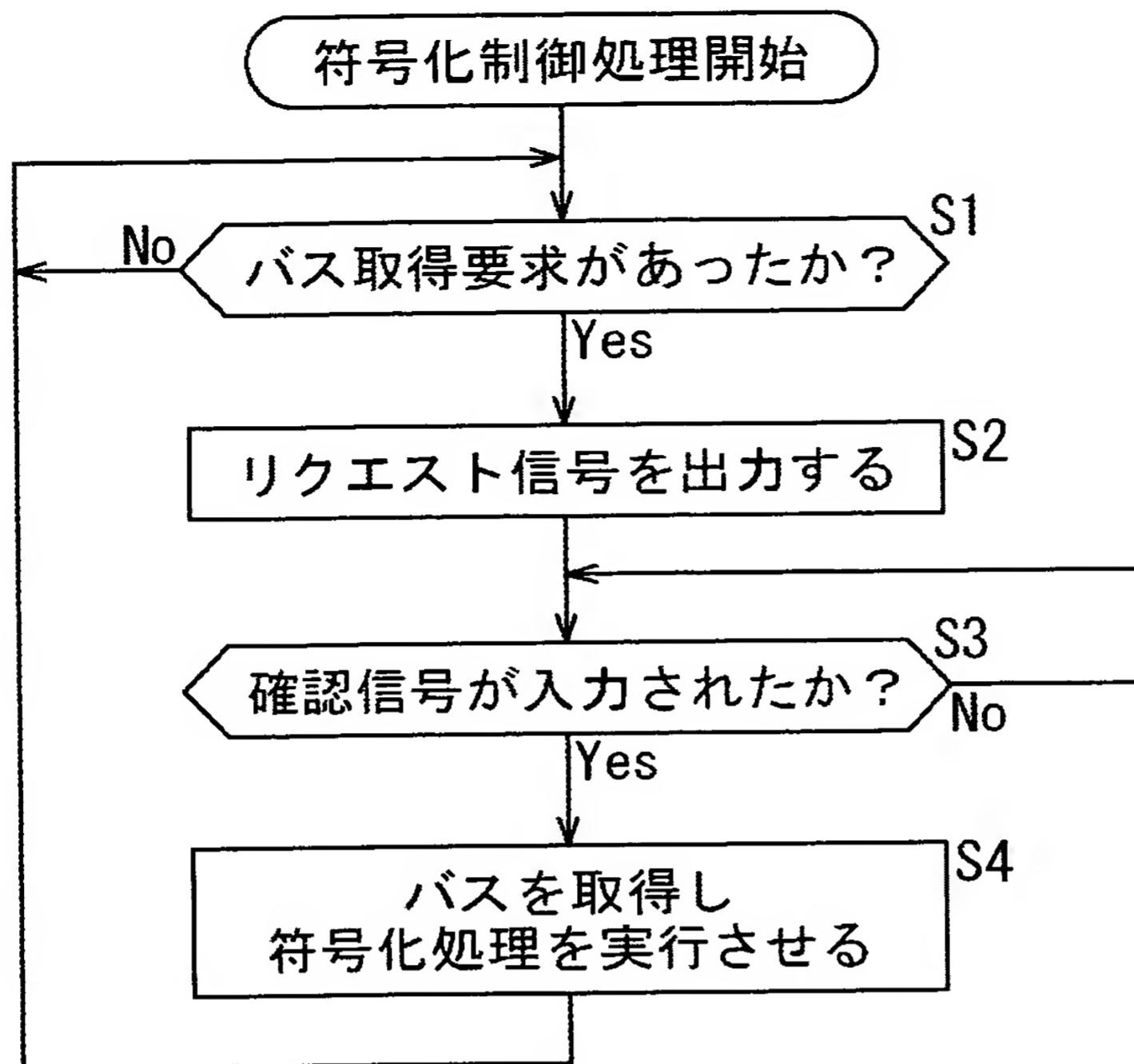


【図 2】



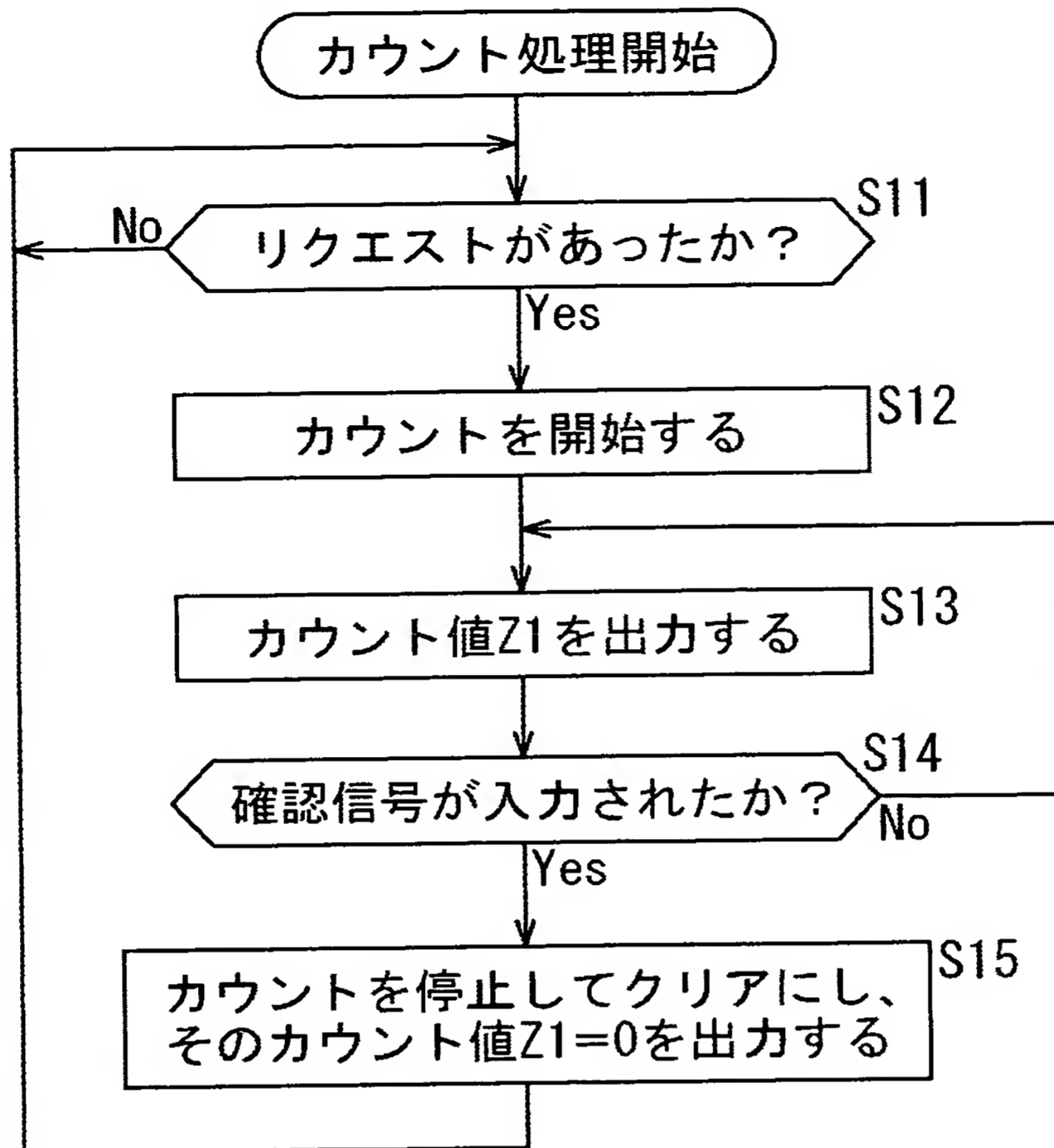
【図 3】

図3



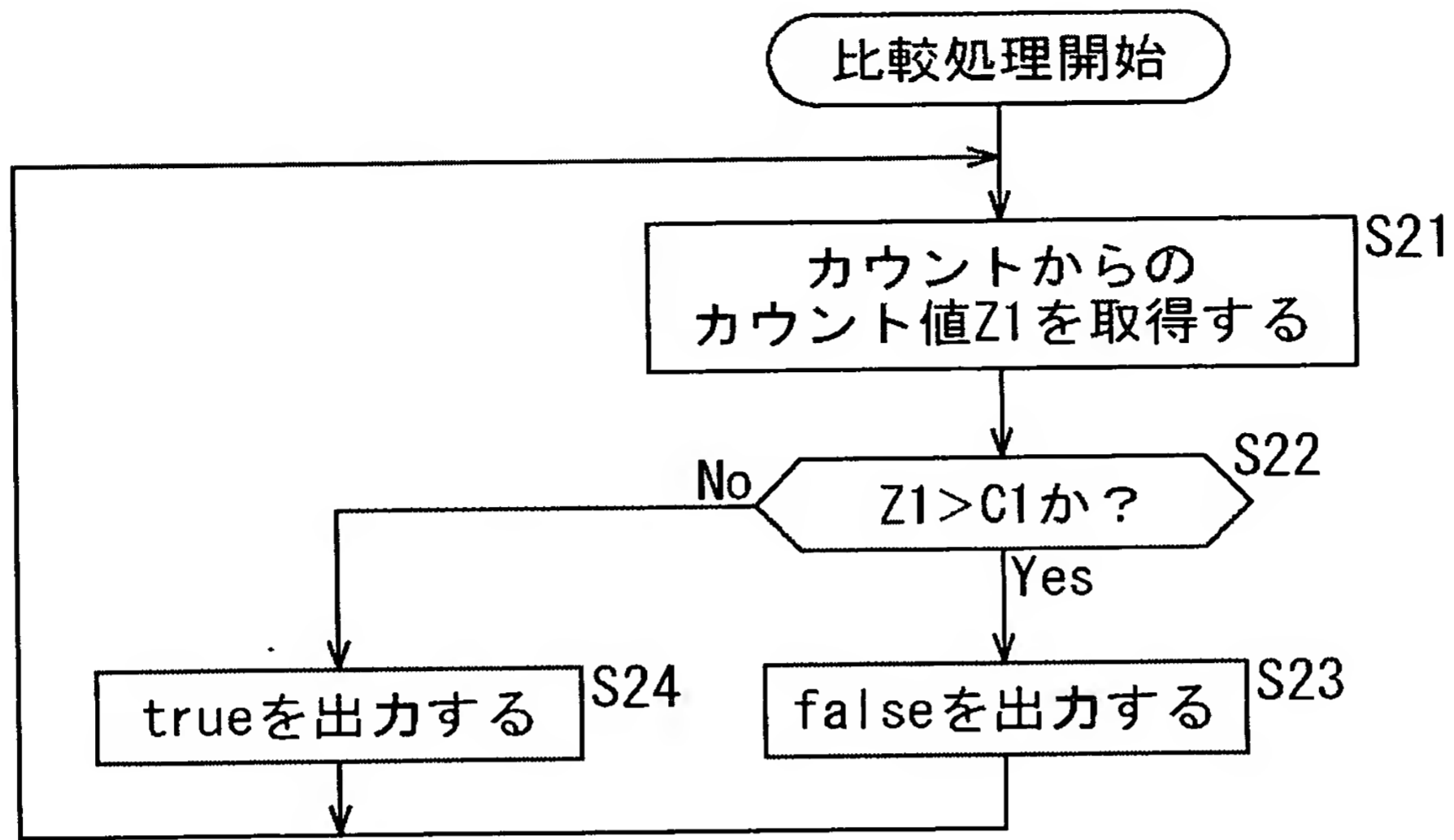
【図 4】

図4



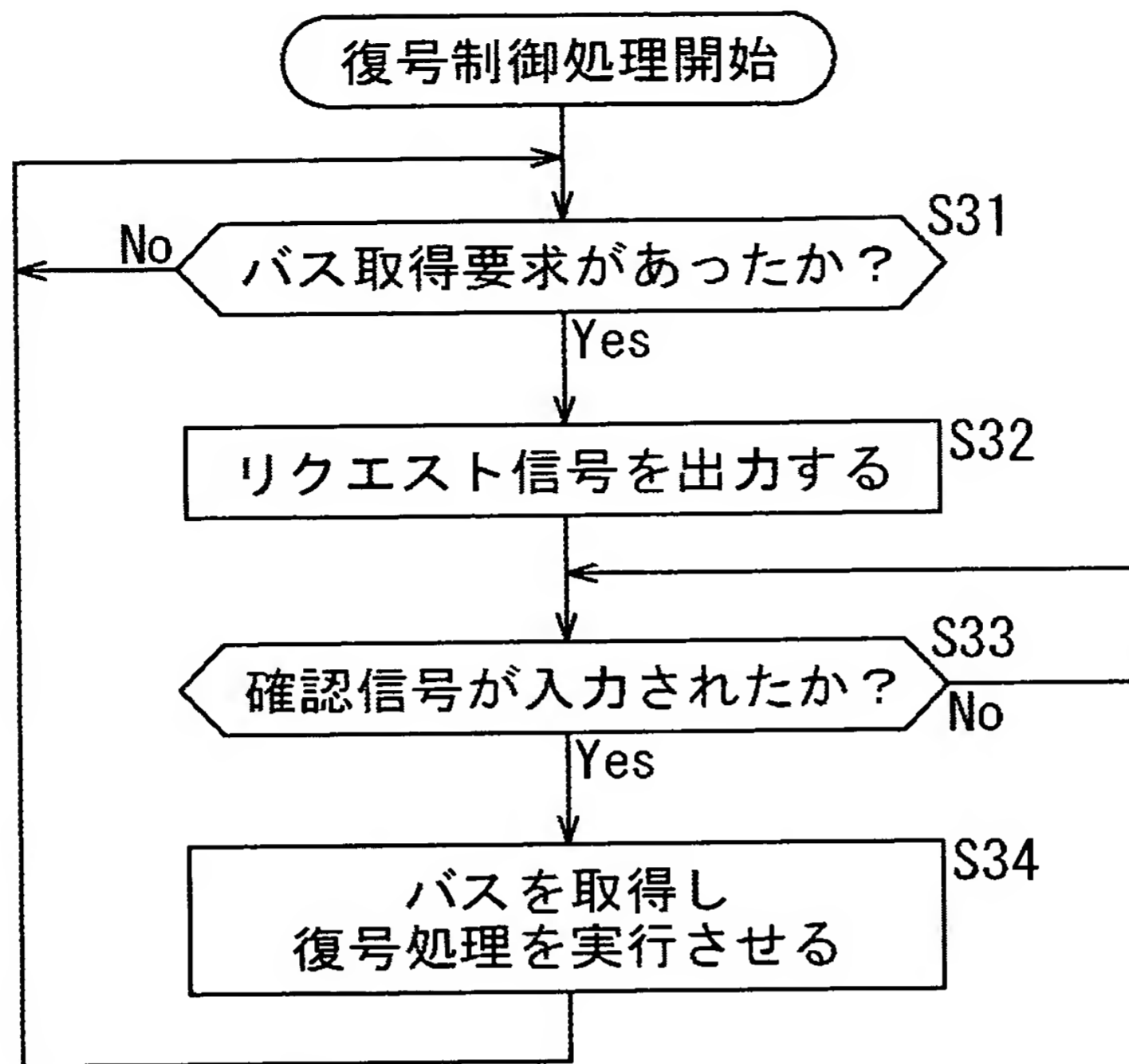
【図 5】

図5



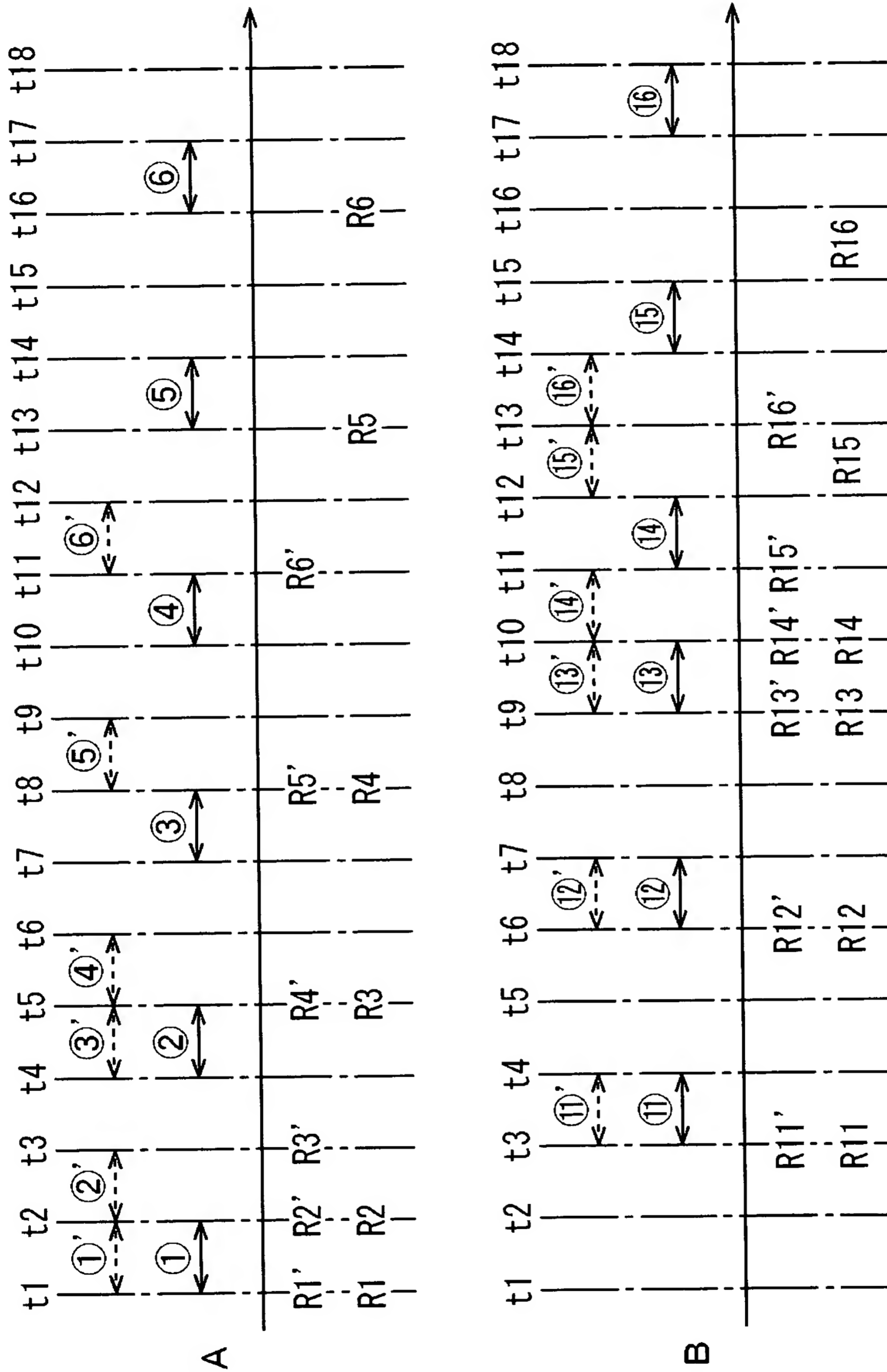
【図 6】

図6



【図 8】

図 8

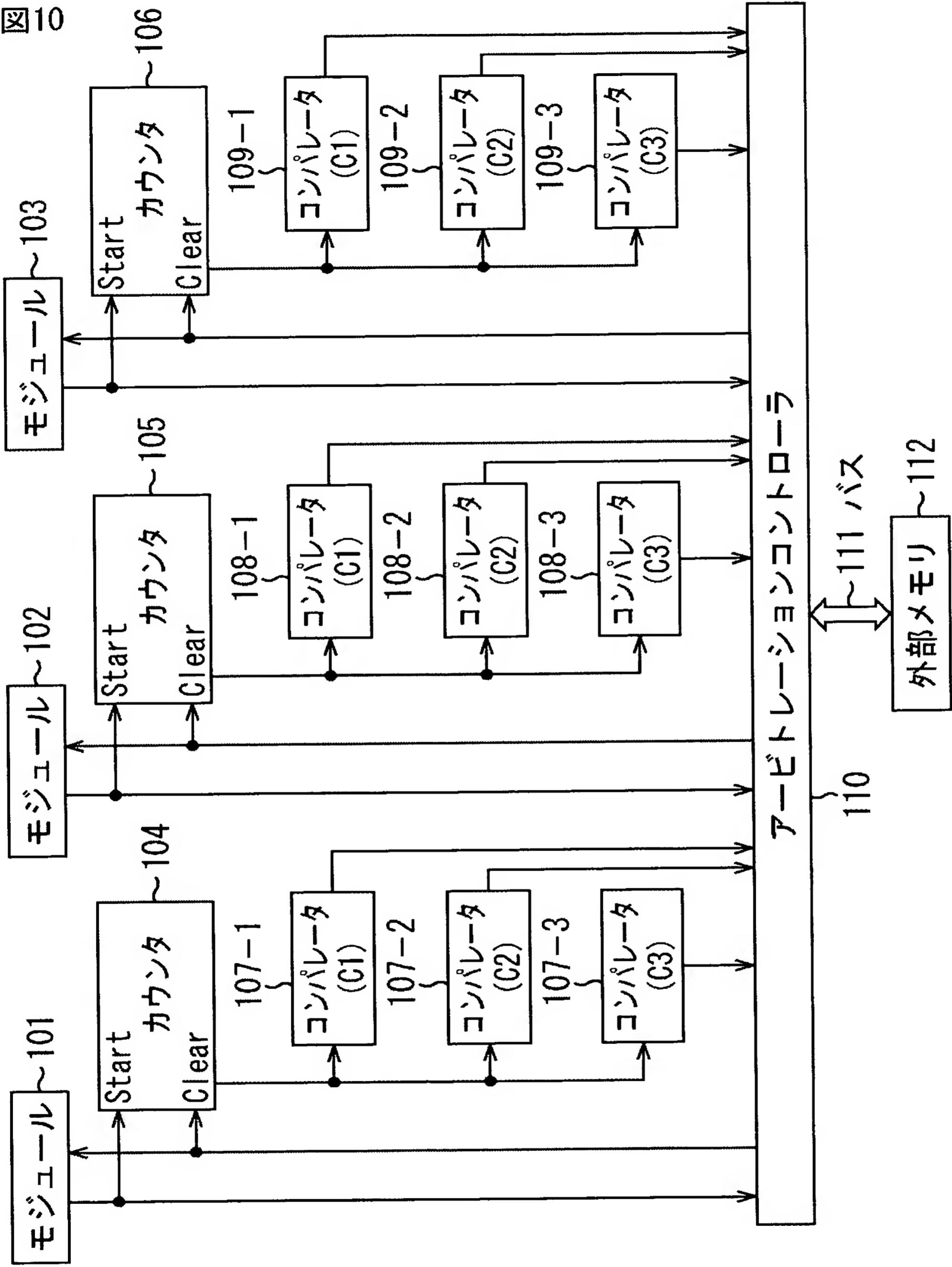


【図 9】

図9

	符号化部 (M0)		復号部 (M1)	
	Comp4-1	Comp4-2	Comp6-1	Comp6-2
R (M0, M1)	0	0	0	0
	1	0	1	0
	1	1	1	1
M0, M1	1	0	0	0
	1	1	0	0
	1	1	1	0
M1, M0	1	0	1	1
	0	0	1	1
	0	0	1	0

【図 10】



【図 1 1】

図11

	M0			M1			M2		
	Comp107-1	Comp107-2	Comp107-3	Comp108-1	Comp108-2	Comp108-3	Comp109-1	Comp109-2	Comp109-3
R (M0, M1, M2)	0	0	0	0	0	0	0	0	0
	1	0	0	1	0	0	1	0	0
	1	1	0	1	1	0	1	1	0
	1	1	1	1	1	1	1	1	1
M0, R (M1, M2)	1	0	0	0	0	0	0	0	0
	1	1	0	0	0	0	0	0	0
	1	1	0	1	0	0	1	0	0
	1	1	1	0	0	0	0	0	0
	1	1	1	1	0	0	1	0	0
	1	1	1	1	1	0	1	1	0
	1	1	1	1	1	0	1	1	0
R (M1, M2), M0	0	0	0	1	0	0	1	0	0
	0	0	0	1	1	0	1	1	0
	0	0	0	1	1	1	1	1	1
	1	0	0	1	1	0	1	1	0
	1	0	0	1	1	1	1	1	1
	1	1	0	1	1	1	1	1	1
	1	1	0	1	0	0	0	0	0
M0, M1, M2	1	1	1	1	0	0	0	0	0
	1	1	1	1	0	0	0	0	0
	1	1	1	1	1	0	0	0	0
	1	1	1	1	1	0	1	0	0
M0, M2, M1	1	1	0	0	0	0	1	0	0
	1	1	1	0	0	0	1	0	0
	1	1	1	0	0	0	1	1	0
	1	1	1	1	0	0	1	1	0

【図 1 2】

図12

	M0			M1			M2		
	Comp107-1	Comp107-2	Comp107-3	Comp108-1	Comp108-2	Comp108-3	Comp109-1	Comp109-2	Comp109-3
M1, R (M0, M2)	0	0	0	1	0	0	0	0	0
	0	0	0	1	1	0	0	0	0
	1	0	0	1	1	0	1	0	0
	0	0	0	1	1	1	0	0	0
	1	0	0	1	1	1	1	0	0
R (M0, M2), M1	1	1	0	1	1	1	1	1	0
	1	0	0	0	0	0	1	0	0
	1	1	0	0	0	0	1	1	0
	1	1	1	0	0	0	1	1	1
	1	1	0	1	0	0	1	1	0
M1, M0, M2	1	1	1	1	0	0	1	1	1
	1	1	1	1	1	0	1	1	1
	1	0	0	1	1	0	0	0	0
	1	0	0	1	1	1	0	0	0
	1	1	0	1	1	1	1	0	0
M1, M2, M0	0	0	0	1	1	0	1	0	0
	0	0	0	1	1	1	1	0	0
	0	0	0	1	1	1	1	1	0
	0	0	0	1	1	1	1	1	0
	1	0	0	1	1	1	1	1	0

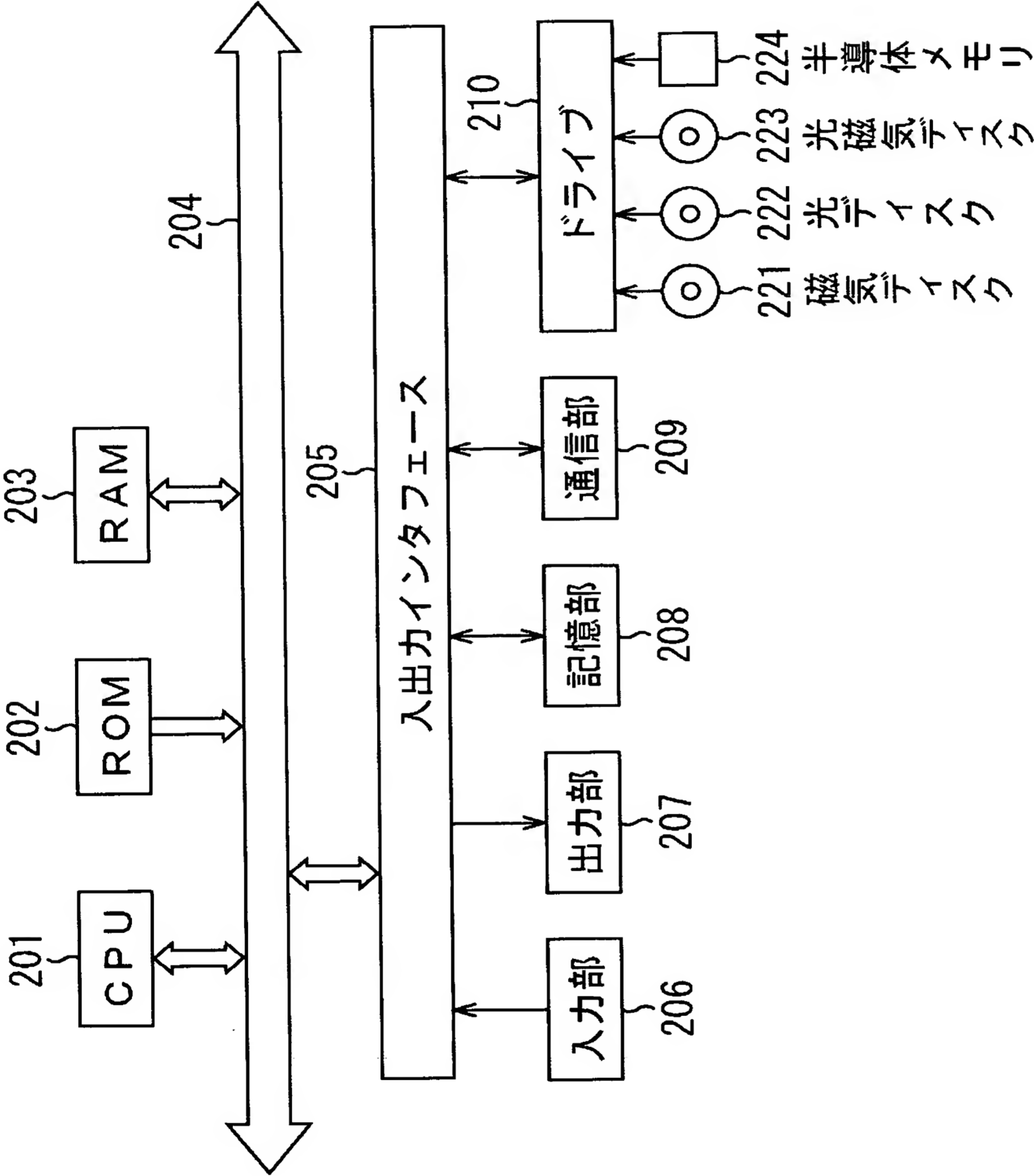
【図 1 3】

図13

	M0			M1			M2		
	Comp107-1	Comp107-2	Comp107-3	Comp108-1	Comp108-2	Comp108-3	Comp109-1	Comp109-2	Comp109-3
M2, R (M0, M1)	0	0	0	0	0	0	1	0	0
	0	0	0	0	0	0	1	1	0
	1	0	0	1	0	0	1	1	0
	0	0	0	0	0	0	1	1	1
	1	0	0	1	0	0	1	1	1
R (M0, M1), M2	1	1	0	1	1	0	1	1	1
	1	0	0	1	0	0	0	0	0
	1	1	0	1	1	0	0	0	0
	1	1	1	1	1	1	0	0	0
	1	1	1	1	1	1	1	1	0
M2, M0, M1	1	0	0	0	0	0	1	1	0
	1	0	0	0	0	0	1	1	1
	1	1	0	0	0	0	1	1	1
	1	1	0	1	0	0	1	1	1
	1	1	0	1	0	0	1	1	0
M2, M1, M0	0	0	0	1	0	0	1	1	0
	0	0	0	1	0	0	1	1	1
	1	0	0	1	1	0	1	1	1
	0	0	0	1	1	0	1	1	1
	0	0	0	1	1	0	1	1	1

【図 14】

図14



【書類名】 要約書

【要約】

【課題】 効率のよいバスアービトレーションを実現させる。

【解決手段】 カウンタ 3, 5 は、それぞれ符号化部 1、および、復号部 2 からリクエスト信号が入力されるとカウントを開始し、カウンタ 3 がカウンタ値 Z 1 をコンパレータ 4 - 1, 4 - 2 に、カウンタ 5 がコンパレータ 6 - 1, 6 - 2 にそれぞれ出力する。コンパレータ 4 - 1, 6 - 1 は、所定の値 C 1 と、コンパレータ 4 - 2, 6 - 2 は、所定の値 C 2 と、それぞれカウンタ値 Z 1, Z 2 を比較し、比較結果をアービトレーションコントローラ 7 に出力する。アービトレーションコントローラ 7 は、このコンパレータ 4 - 1, 4 - 2, 6 - 1, 6 - 2 から入力される信号に基づいて、符号化部 1 と復号部 2 のモジュールの優先順位を決定し、最も優先順位の高いモジュールに確認信号を出力する。本発明は、LSI に適応することができる。

【選択図】 図 2

特願 2 0 0 2 - 2 8 9 0 2 2

出 願 人 履 歴 情 報

識別番号 [0 0 0 0 0 2 1 8 5]

1. 変更年月日	1 9 9 0 年 8 月 3 0 日
[変更理由]	新規登録
住 所	東京都品川区北品川 6 丁目 7 番 3 5 号
氏 名	ソニー株式会社